



AWS Certified Developer (Associate) Crash Course

Course Intro and
AWS Cloud Overview

Certification Tracks

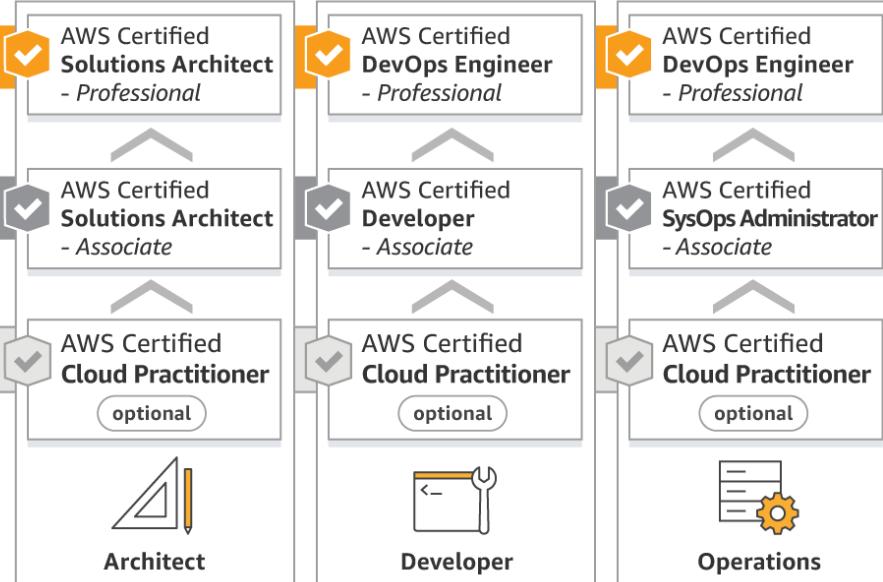


Professional

Associate

Foundational

Role-Based Certifications



Specialty Certifications



Specialty Certification requires
Cloud Practitioner or
Associate-level certification

Certification Tracks



Course Overview

GOAL:



Certified

Developer - Associate

AWS Developer Associate Blueprint

<https://aws.amazon.com/certification/certified-developer-associate/>

Recommended AWS Knowledge (for the test)

- **One or more years of hands-on experience** developing and maintaining an AWS based application
- In-depth knowledge of at least one high-level programming language
- Understanding of core AWS services, uses, and basic AWS architecture best practices

AWS Developer Associate Blueprint

<https://aws.amazon.com/certification/certified-developer-associate/>

Recommended AWS Knowledge (for the test)

- One or more years of hands-on experience

**This course is not a substitute
for real-world experience.**

We'll be moving very quickly.

AWS Developer Associate

Recommended AWS Knowledge – Continued

- Proficiency in developing, deploying, and debugging cloud-based applications using AWS
- Ability to use the AWS service APIs, AWS CLI, and SDKs to write applications
- Ability to identify key features of AWS services
- **Understanding of the AWS shared responsibility model**

AWS Developer Associate

Recommended AWS Knowledge – Continued

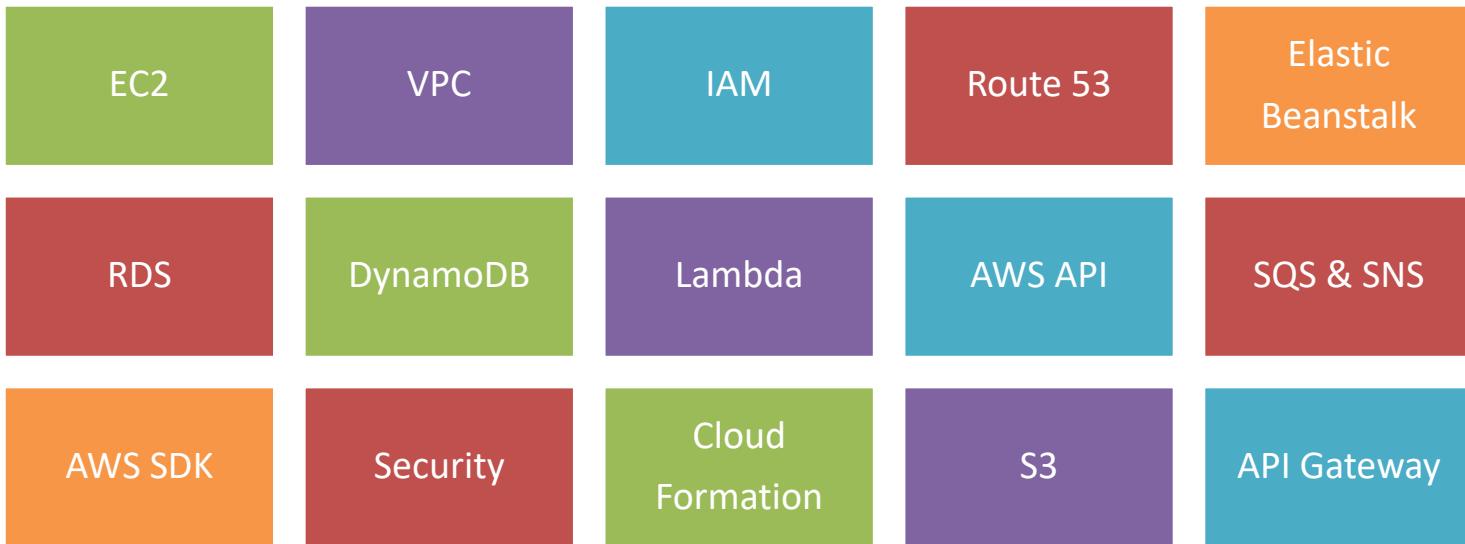
- Understanding of application lifecycle management
- Ability to use a CI/CD pipeline to deploy applications on AWS
- Ability to use or interact with AWS services
- Ability to apply a basic understanding of cloud-native applications to write code
- Ability to write code using AWS security best practices (e.g., not using secret and access keys in the code, instead using IAM roles)

AWS Developer Associate

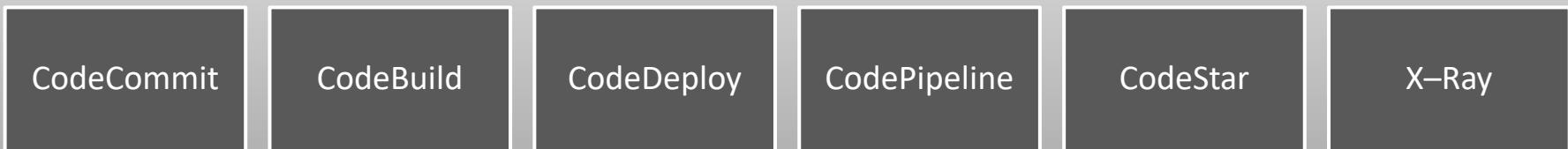
Recommended AWS Knowledge – Continued

- Proficiency writing code for serverless applications
- Understanding of the use of containers in the development process

AWS Developer Associate Topics



AWS Developer Tools



Course Prerequisites

- Fundamentals of the various AWS Services
- Experience interacting with AWS services
 - CLI
 - Console
- Command-line Understanding

Helpful:

- AWS Account Access
- SSH Client

What not to expect

The guidelines of the AWS Certification Program Agreement apply to this course.

2. Testing.

2.1. Testing Rules.

You will not engage in any misconduct in connection with the Certification Exam, including:

<...>

(f) disclosing or disseminating the content of any Certification Exam or Testing Materials

- You will not see test questions in this course.
- You will not hear, "...expect to see this on the test..."

<https://aws.amazon.com/certification/certification-agreement/>

AWS Developer Associate

Exam Overview

- **Multiple-choice:** Has one correct response and three incorrect responses (distractors).
- **Multiple-response:** Has two or more correct responses out of five or more options.
- Pass or Fail
- Score: 100 – 1000, Minimum: 720

AWS Developer Associate

Five Domains

- Domain 1: Deployment 22%
- Domain 2: Security 26%
- Domain 3: Development with AWS Services 30%
- Domain 4: Refactoring 10%
- Domain 5: Monitoring and Troubleshooting: 12%

AWS Developer Associate

Domain 1: Deployment

- 1.1** Deploy written code in AWS using existing CI/CD pipelines, processes, and patterns.
- 1.2** Deploy applications using Elastic Beanstalk.
- 1.3** Prepare the application deployment package to be deployed to AWS.
- 1.4** Deploy serverless applications.

AWS Developer Associate

Domain 2: Security

- 2.1** Make authenticated calls to AWS services.
- 2.2** Implement encryption using AWS services.
- 2.3** Implement application authentication and authorization.

AWS Developer Associate

Domain 3: Development with AWS Services

3.1 Write code for serverless applications.

3.2 Translate functional requirements into application design.

3.3 Implement application design into application code.

3.4 Write code that interacts with AWS services by using APIs, SDKs, and AWS CLI.

AWS Developer Associate

Domain 4: Refactoring

- 4.1** Optimize application to best use AWS services and features.
- 4.2** Migrate existing application code to run on AWS.

Domain 5: Monitoring and Troubleshooting

- 5.1** Write code that can be monitored.
- 5.2** Perform root cause analysis on faults found in testing or production.

Author



Nick Garner

- Denali Advanced Integration
- Cisco Advanced Services, 10 Years
- Prior to Cisco, 10 years in financial and military verticals.
- Network Design, Security Architecture
- Cisco Press Technical Editor
- Pearson Video Author
- AWS SA, Dev
- CCIE R/S and Security #17871
- CISSP / CEH

Poll Audience

Which continent are you joining from?

- Africa
- Antarctica
- Asia
- Europe
- North America
- South America
- Australia/Oceania

Poll Audience

What is your experience with AWS?

- What's AWS?
- I've launched a few instances.
- I administer/operate an existing application running on AWS.
- I've created and deployed an entire application running in AWS.
- I should be teaching this class!

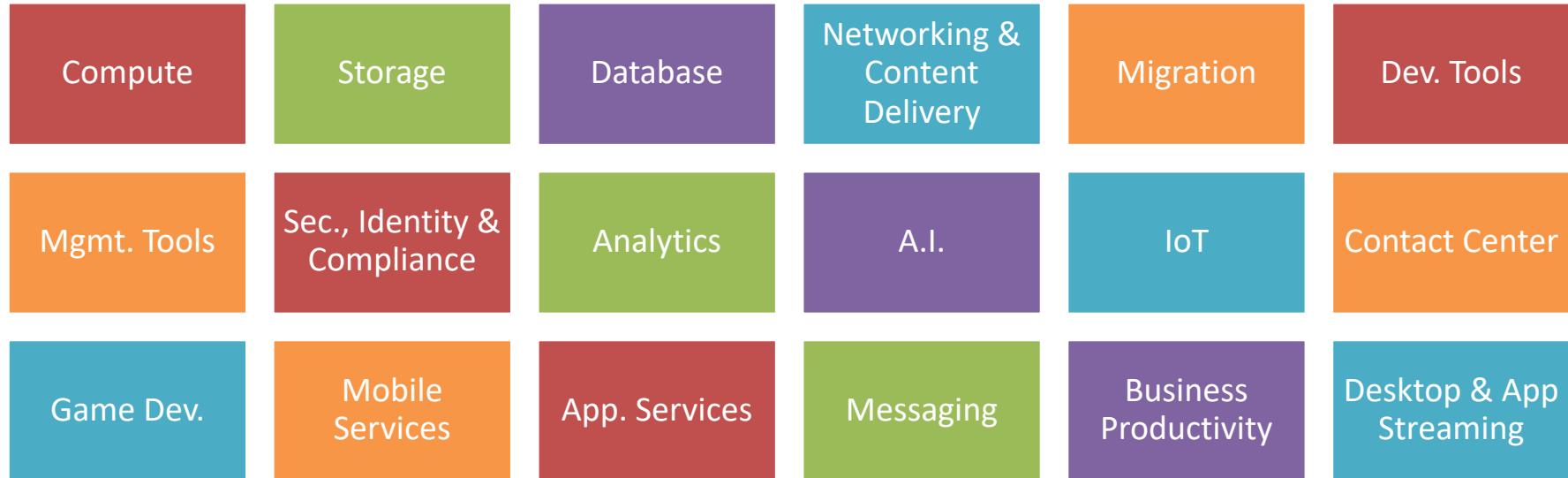


AWS Certified Developer (Associate) Crash Course

AWS Cloud Overview

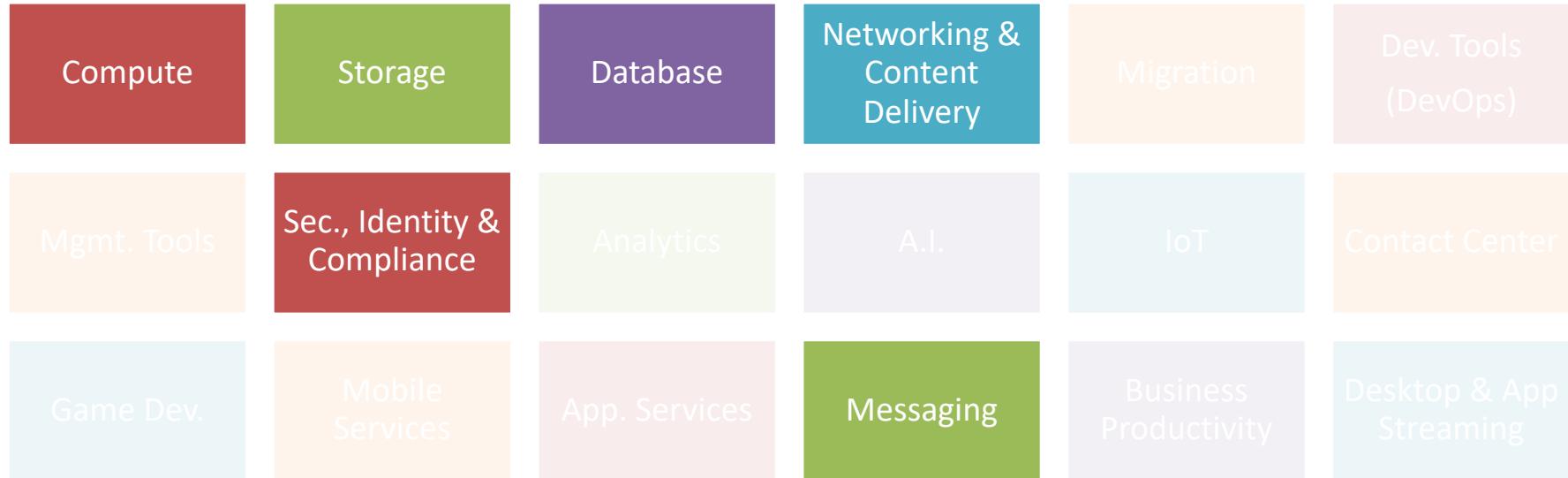
AWS Services Overview

- AWS Services Divided into Function Buckets



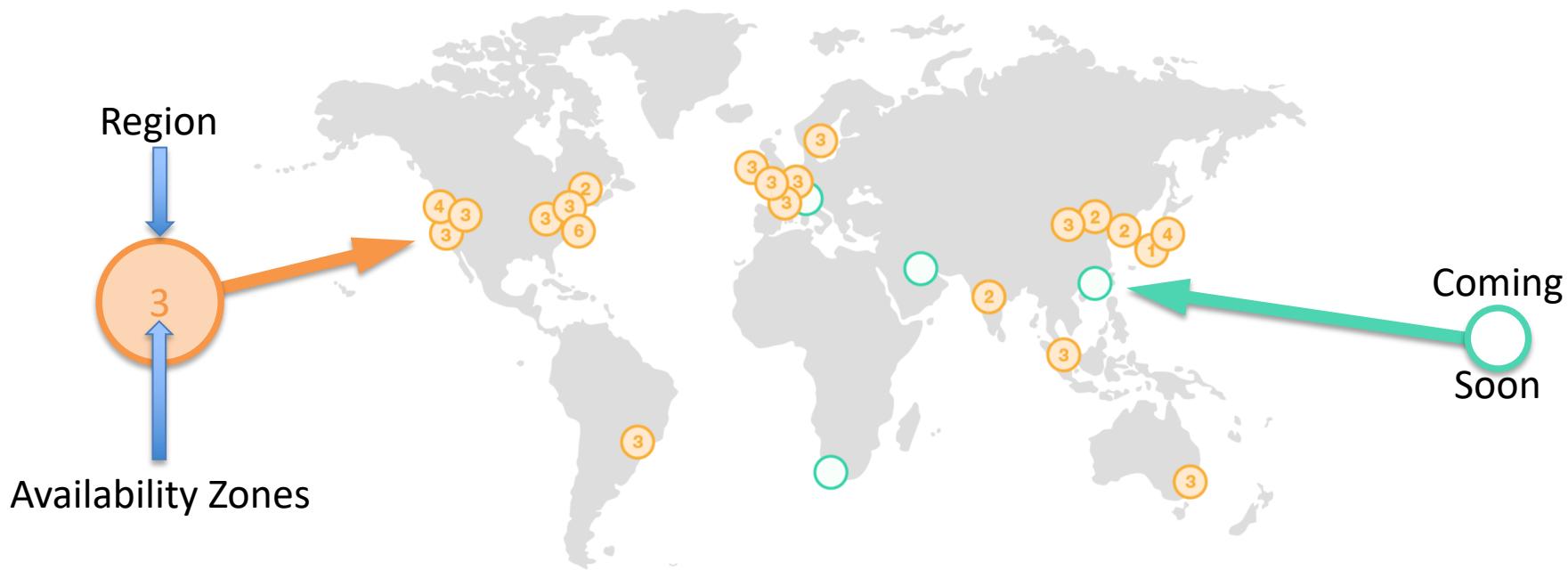
AWS Services Overview

Developer Areas of Focus

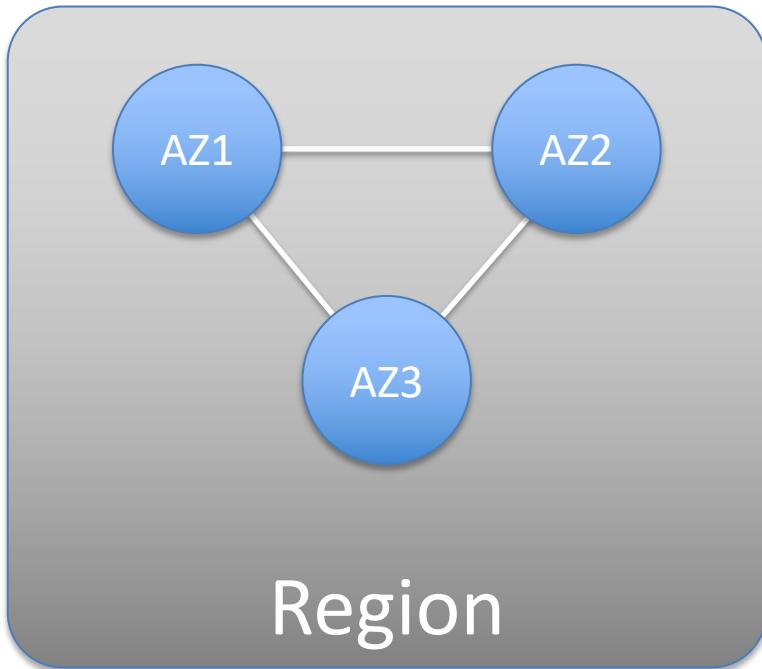


AWS Regions

AWS is Globally Distributed



Regions vs. Availability Zones



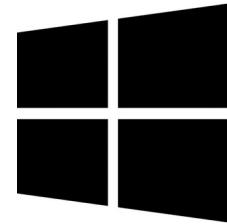
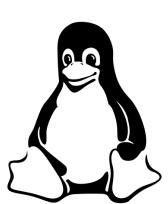
- Region:
 - Completely Independent Entity
 - AWS Console View is Region Constrained
- Availability Zones:
 - Intra-Region Fault Isolation
 - Cluster of Data Centers
 - Connected to other AZ via low latency links
 - Many AZs have dedicated power substations.



AWS Certified Developer (Associate) Crash Course

AWS CLI Installation & Usage

AWS CLI Installation & Usage

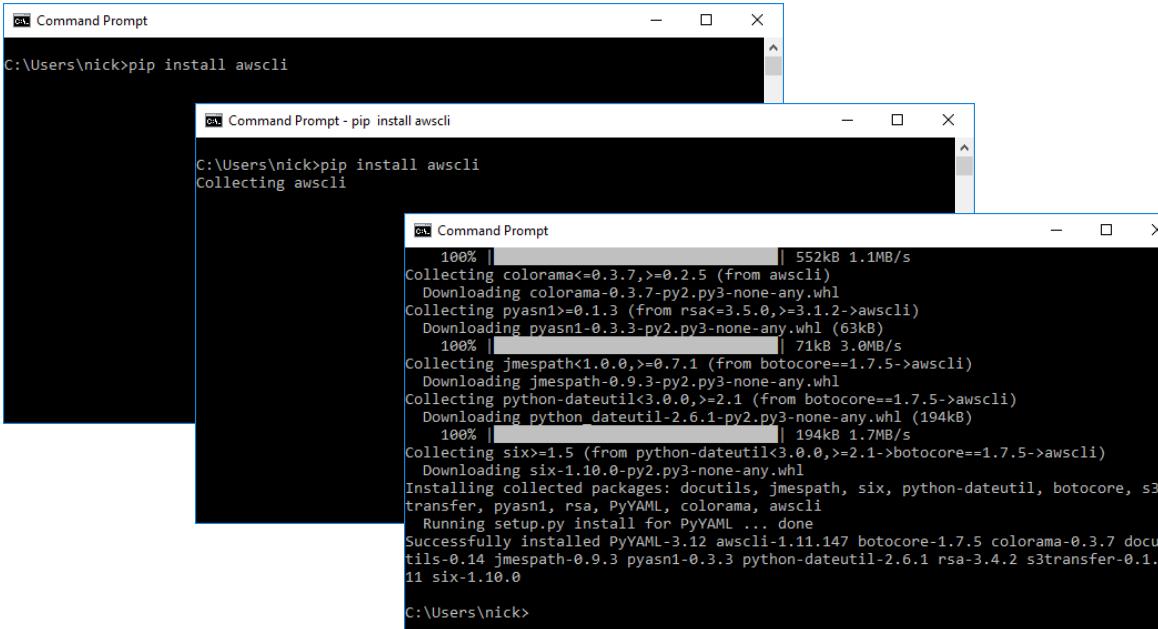


- Python & pip
- Requires Python >= 2.6.5
- 64 & 32-bit Installer Packages
- Pip works too

```
pip install awscli
```

AWS CLI Installation & Usage

AWS CLI Installation using pip on Windows



AWS CLI Installation & Usage

AWS CLI Installation using pip on WSL

The screenshot shows a terminal window with two tabs. The left tab displays the output of the command `sudo apt install python python-pip`, which installs various dependencies and the AWS CLI package. The right tab shows the command `pip install awscli` being run, which installs the AWS CLI library. Both processes are shown with progress bars and file names.

```
nick@NUC:~$ sudo apt install python python-pip
[sudo] password for nick:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  libfreetype6
Use 'sudo apt autoremove' to remove it.
The following packages will be installed:
  binutils binutils-common binutils-x86_64-linux-gnu build-essential cpp cpp-7 dkp
  gcc-7-base libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-p
  libc-dev-bin libc6-dev libcc1-0 libcurlkrts5 libdpkg-perl libexpat1-dev libfakenc
  libgcc-7-dev libgomp1 libis119 libitm1 libisane libmpc3 libmpx2 libpython-all-de
  libpython2.7 libpython2.7-dev libpython2.7-minimal libpython2.7-stdlib libquadma
  libubsan0 libxml2-dev make manpages-dev python-all python-all-dev python-asn1
  python-crypto python-cryptography python-dbus python-dev python-enums4 python-gi
  python-keyring python-keyrings.alt python-minimal python-pip-whl python-pkg-resc
  python-setuptools python-six python-wheel python-xdg python2.7 python2.7-dev pyt
  suggested packages:
  binutils-doc cpp-doc gcc-7-locales debian-keyring g++-multilib g++-7-multilib gc
  gcc-7-base libautoconf automake libtinfo flex bison gdb gcc-doc gcc-7-mudflap lib
  libatomic1-dev libasan4-dev liblsan4-dev libubsan0-dev libcurlkrts5
  glibc-doc bzr libstdc++-7-doc make-doc python-doc python-tk python-crypto-doc py
  python-cryptography-vectors python-dbus-doc python-dbus-doc python-enums4-doc py
  libcurl4 libcurl4-openssl-dev libpython2.7 python2.7 python2.7-fs python-gdata python-keyczar p
  python-setuptools-doc python2.7-doc binfmt-support
The following NEW packages will be installed:
  binutils binutils-common binutils-x86_64-linux-gnu build-essential cpp cpp-7 dkp
  gcc-7-base libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-p
  libc-dev-bin libc6-dev libcc1-0 libcurlkrts5 libdpkg-perl libexpat1-dev libfakenc
  libgcc-7-dev libgomp1 libis119 libitm1 libisane libmpc3 libmpx2 libpython-all-de
  libpython2.7 libpython2.7-dev libpython2.7-minimal libpython2.7-stdlib libquadma
  libubsan0 libxml2-dev make manpages-dev python python-all python-all-dev python
  python-crypto python-cryptography python-dbus python-dev python-enums4 python-gi
  python-keyring python-keyrings.alt python-minimal python-pip python-pip-whl python
  python-secretstorage python-setuptools python-six python-wheel python-xdg python2.7-dev python2.7
  0 upgraded, 75 newly installed, 0 to remove and 0 not upgraded.
Need to get 74.2 MB of archives.
After this operation, 235 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

```
nick@NUC:~$ pip install awscli
Collecting awscli==1.12.122
  Downloading https://files.pythonhosted.org/packages/62/00/e1d7de624db8ba7090d1226aebefab96a2c71cd5cf
    100% |████████████████████████████████| 235kB 394kB/s
Collecting s3transfer<0.1.3,>=0.1.2 (from awscli==1.12.122)
  Downloading https://files.pythonhosted.org/packages/7b/7c/c9386b82a25115cccf1903441bba3cbadcfae7b678a
    100% |████████████████████████████████| 73kB 122kB 564kB/s
Collecting pyasn1<0.1.3,>=0.1.2 (from awscli==1.12.122)
  Downloading https://files.pythonhosted.org/packages/7b/fb/00a976f728d01fecfe898238ce23f502a721c0ac0e
    100% |████████████████████████████████| 73kB 81kB 345kB/s
Collecting futures<4.0.0,>=2.2.0; python_version == "2.7" (from awscli==1.12.122)
  Downloading https://files.pythonhosted.org/packages/2d/99/b2c4e9d5a30f6471e410a146232b4118e697fa3ffcc
    100% |████████████████████████████████| 606kB 606a65efde84deb0/futures-3.2.0-py2-none-any.whl
Collecting six<1.5,>=1.4.9 (from awscli==1.12.122)
  Downloading https://files.pythonhosted.org/packages/73/fb/00a976f728d01fecfe898238ce23f502a721c0ac0e
    100% |████████████████████████████████| 17kB 17kB 17kB/s
Collecting botocore==1.12.122 (from awscli==1.12.122)
  Downloading https://files.pythonhosted.org/packages/62/00/e1d7de624db8ba7090d1226aebefab96a2c71cd5cf
    100% |████████████████████████████████| 235kB 394kB/s
Collecting s3transfer<0.1.3,>=0.1.2 (from botocore==1.12.122->awscli)
  Downloading https://files.pythonhosted.org/packages/7b/7c/c9386b82a25115cccf1903441bba3cbadcfae7b678a
    100% |████████████████████████████████| 73kB 122kB 564kB/s
Collecting pyasn1<0.1.3,>=0.1.2 (from botocore==1.12.122->awscli)
  Downloading https://files.pythonhosted.org/packages/7b/fb/00a976f728d01fecfe898238ce23f502a721c0ac0e
    100% |████████████████████████████████| 73kB 81kB 345kB/s
Building wheels for collected packages: PyYAML
  Running setup.py bdist_wheel for PyYAML ... done
  Stored in directory: /home/nick/.cache/pip/wheels/ad/da/0c/74eb680767247273e2cf2723482cb9c924fe70af57
  c334513f
Successfully built PyYAML
Installing collected packages: docutils, jmespath, six, python-dateutil, urllib3, botocore, PyYAML, pyasn1, rsa, colorama, futures, s3transfer, awscli
Successfully installed PyYAML-3.13 awscli-1.16.132 botocore-1.12.122 colorama-0.3.9 docutils-0.14 futures-3.2.0 jmespath-0.9.4 pyasn1-0.4.5 python-dateutil-2.8.0 rsa-3.4.2 s3transfer-0.2.0 six-1.12.0 urllib3-1.24.1
nick@NUC:~$
```

AWS CLI Setup

CLI Demo



AWS Certified Developer
(Associate) Crash Course
Foundational Services

Agenda

- Shared Responsibility Model
- IAM
- VPC
- EC2
- Route53

A large, light-gray circular icon containing a white play triangle, positioned on the left side of the slide.

AWS Certified Developer (Associate) Crash Course

Foundational Services

Shared Responsibility Model

AWS Shared Responsibility Model

Security *of* the Cloud

vs.

Security *in* the Cloud

AWS Shared Responsibility Model

Security and Compliance:

Shared between AWS and the customer

AWS Manages:

- Host OS
- Virtualization
- Physical Security
- Relieves your operational burden

AWS Shared Responsibility Model

You are responsible for:

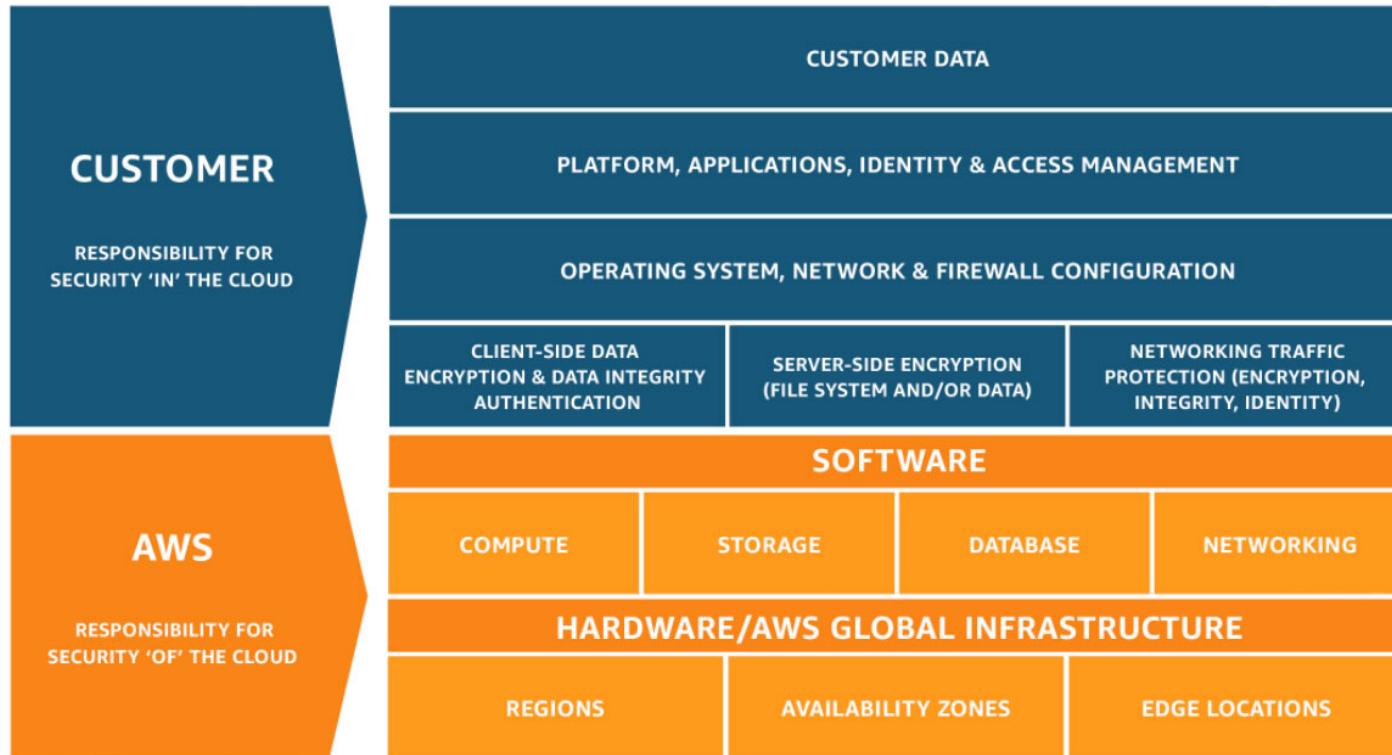
- Management of the guest operating system (updates / patches) in EC2
- Application software
- Rule configuration on AWS provided firewall
- Laws and regulations surrounding your chosen services

AWS Shared Responsibility Model

Shared Responsibilities

- Patch Management
 - AWS is responsible for patching infrastructure
 - You are responsible for patching the guest OS and applications
- Configuration Management
 - AWS maintains configuration of infrastructure devices
 - You are responsible for configuring the guest operating systems, databases, and applications
- Awareness & Training
 - AWS trains AWS employees,
 - You must train your own employees

AWS Shared Responsibility Model





AWS Certified Developer (Associate) Crash Course

Foundational Services

IAM

IAM Overview

Identity

Who are you?

Access

What can you do?

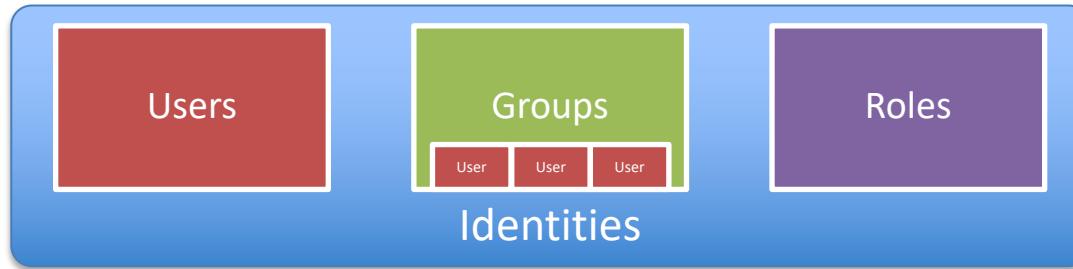
Management

Administration of...

- AWS Service Permissions
 - Federated Identity
 - Free
- Fine-grained Permissions
 - Multi-factor Authentication
 - PCI Compliant
 - Replicated Worldwide

IAM Overview

IAM Components



IAM Overview

Users

- An individual entity with a defined username.

Access Types:

- Programmatic Access
- AWS Management Console Access

NOTE: The account you initially create the AWS account with is the “root” user. Account has full access and should be secured.

Policies – JSON Document

- A set of **permissions** **Created by:**
 - Effect
 - Action
 - Resource
 - Condition
 - Copy of AWS Policy
 - Policy Wizard
 - Self-defined

IAM Overview

Groups

- A collection of **Users**
- Defined by a **Group Name**
 - Group name can be changed at any time
However, don't do this, ARN will change.
- Have a **Policy** attached

IAM Overview

Roles

- AWS identity with permission policies
- Can be assumed by anyone/anything that needs it and with the necessary permissions granted.

Use

- Delegate access to users, applications, or services that don't normally have access to your AWS resource

IAM Overview

AWS Security Token Service

- Create/provide trusted users with temporary security credentials -> access your AWS resources.
- Almost identical to long-term creds:
 - STS creds are *short-term*.
 - Configurable expiration, minutes to hours.
 - Expired? Unknown, denied.
 - Dynamically generated upon request.
 - User/app can request new creds before expiration.

IAM Overview

AWS Security Token Service

Advantages:

- No need to distribute creds with applications.
- Provide access without requiring an AWS identity (user)
- Basis for roles and identity federation
- Credentials expire, no need to rotate/distribute new creds.

IAM Overview

AWS Security Token Service

Works with CLI:

```
$ aws sts assume-role \
  --role-arn arn:aws:iam::123456789012:role/role-name \
  --role-session-name "RoleSession1" \
  --profile IAM-user-name > assume-role-output.txt
```

```
$ export AWS_ACCESS_KEY_ID=AKIAI44QH8DHBEXAMPLE
$ export AWS_SECRET_ACCESS_KEY=<cut>
$ export AWS_SESSION_TOKEN=AQoDYXdzEJr...<cut>
$ aws ec2 describe-instances --region us-west-1
```

IAM Overview

AWS Security Token Service

Expiration Limits:

API GetFederationToken:

Minimum: 900 seconds / 15 minutes

Maximum: 129,600 / 36 hours

Default: 43,200 seconds / 12 hours

API GetSessionToken:

Minimum: 900 seconds / 15 minutes

Maximum: 129,600 / 36 hours

Default: 43,200 seconds / 12 hours

IAM Overview

Identity Providers (IdP)

- Integrate external identity database
- Can assign permissions to users in that external IdP
- Example: Corporate User Directory

Compatible IdPs

- OpenID Connect (OIDC)
- Security Assertion Markup Language 2.0 (SAML)

IAM Overview

Web Identity Federation

Example Use Case:

Mobile game that stores player and score information in S3 and DDB.

Requires: a method to authenticate to S3/DDB.

Best Practice: Don't embed keys in code.

Solution: Use temporary creds that map to a role with S3/DDB policies.

The Easy Way: Amazon AWS Cognito

Amazon Cognito Federated Identities

- “Access control for your resources”
- Control access to your AWS resources and APIs.
- Map users to different roles and permissions and get temporary AWS credentials for
 - E.g. Amazon S3, Amazon DynamoDB, Amazon API Gateway, and AWS Lambda

IAM Overview

Amazon Cognito User Pools

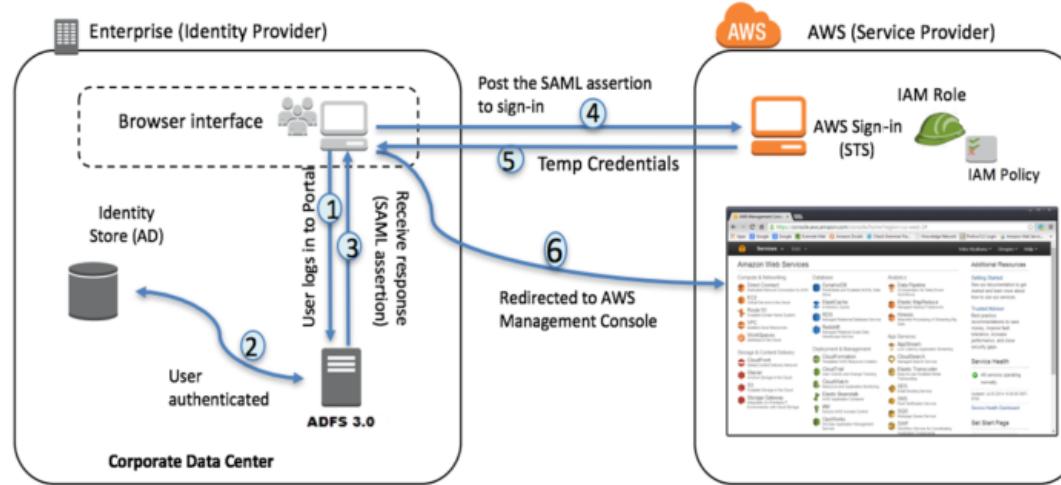
- **“A directory for all your users”**
- Directory to sign up and sign in users,
- Store user profiles
- Customizable User Interface (match your app)
- Integration with social identity providers
 - E.g. Facebook, Google, and Amazon, MS Active Directory through SAML, etc.

Amazon Cognito

- Pricing per monthly active user (MAU)
 - First 50k users, free.
- <https://aws.amazon.com/cognito/>

IAM Overview

AWS Federated Authentication with Active Directory Federation Services (AD FS)



ADFS is an SSO solution created by Microsoft.

<https://aws.amazon.com/blogs/security/aws-federated-authentication-with-active-directory-federation-services-ad-fs/>

IAM Overview

IAM Demo



AWS Certified Developer (Associate) Crash Course

Foundational Services

VPC

VPC Overview

Virtual

Exists, can't touch it

Private

Segregated

Cloud

Looks like your DC
but not in it

VPC Overview

- Virtual network dedicated to your AWS account
- Logically isolated from other VPCs
- Some AWS Services are *launched into* or *attached to* a VPC
- Configurable:
 - Subnet Ranges
 - Routing Tables
 - Gateways
 - Security

VPC Overview

Networking

- Public & private addresses
- Preserved across reboots
- Multiple IPs can be assigned to an instance
- Define and attach multiple network interfaces to one instance
- IPv6 supported

Security

- Security Groups
- Network ACLs
- Reassign groups on the fly
- Assign groups to VPC and EC2 instances
- Ingress & egress filtering

VPC Overview

Default VPC

- EC2-VPC, different from EC2 Classic
- Default VPC is created when account created
- Default VPC has default subnet in each availability zone
- Instance can be created without any knowledge of VPC
(Don't need to assign subnet)

Non-default VPC

- Any VPC you create manually
- Behaves the same as default VPC

VPC Overview

Internet Access

- Access from VPC to Internet is via Internet Gateway (IGW)
- Access to/from Instance with public IP is via IG
- Instances with only Private IP address cannot access Internet but can talk to each other
- Alternate: NAT Gateway through IG for instances with only private IP addresses
- **Default:** VPC has no gateways.

VPC Overview

Example AWS Services that can be launched into a VPC:

AWS
Data Pipeline

EC2

Elastic
Beanstalk

Elastic
Load Balancing

ElastiCache

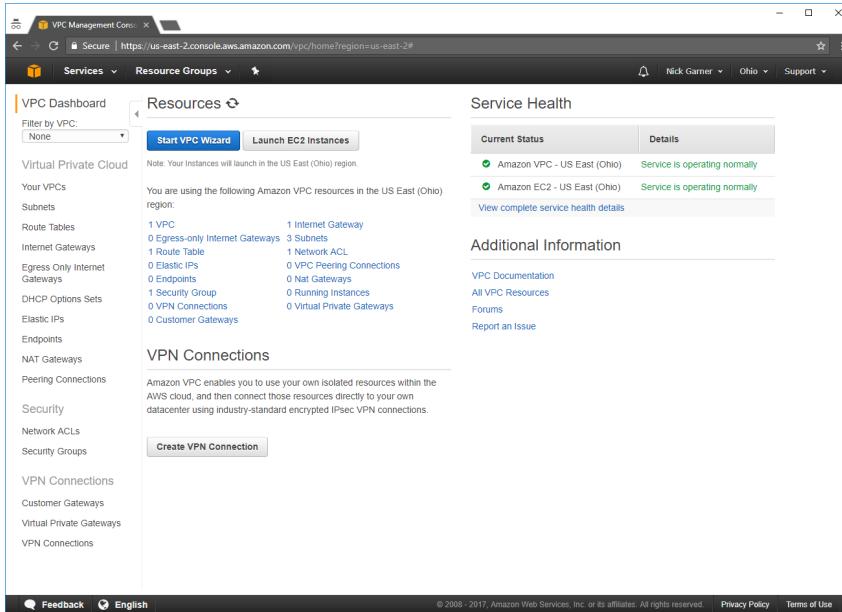
AWS
OpsWorks

RDS

Redshift

Workspaces

VPC Overview



Accessing/Managing VPCs

- Web UI
- AWS Command Line Interface
- AWS Tools for Windows PowerShell
- AWS Query API (REST)

VPC Overview

Pricing

- VPC: Free
- Charges associated with:
 - EC2 Instances
 - NAT Gateways
 - VPN Connections

<http://aws.amazon.com/ec2/pricing/>
<http://aws.amazon.com/vpc/pricing/>

Soft* Limits (July 2020)

- VPC/Region: 5
- Subnets/VPC: 200
- Elastic IP/Region: 5
- Inet Gateways/Region: 5
- Network ACLs/VPC: 200
- Rules/Network ACL: 20

Google: “aws vpc limits”

* You can request an increase to certain limits.

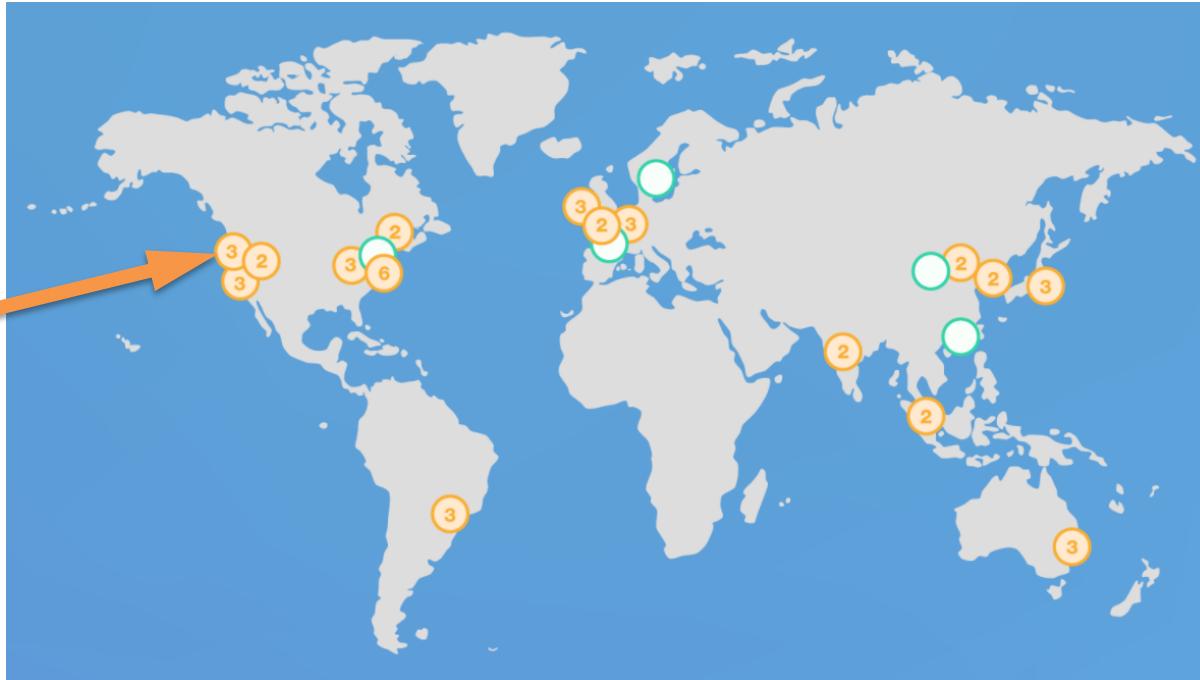
AWS Regions Reminder

AWS is Globally Distributed

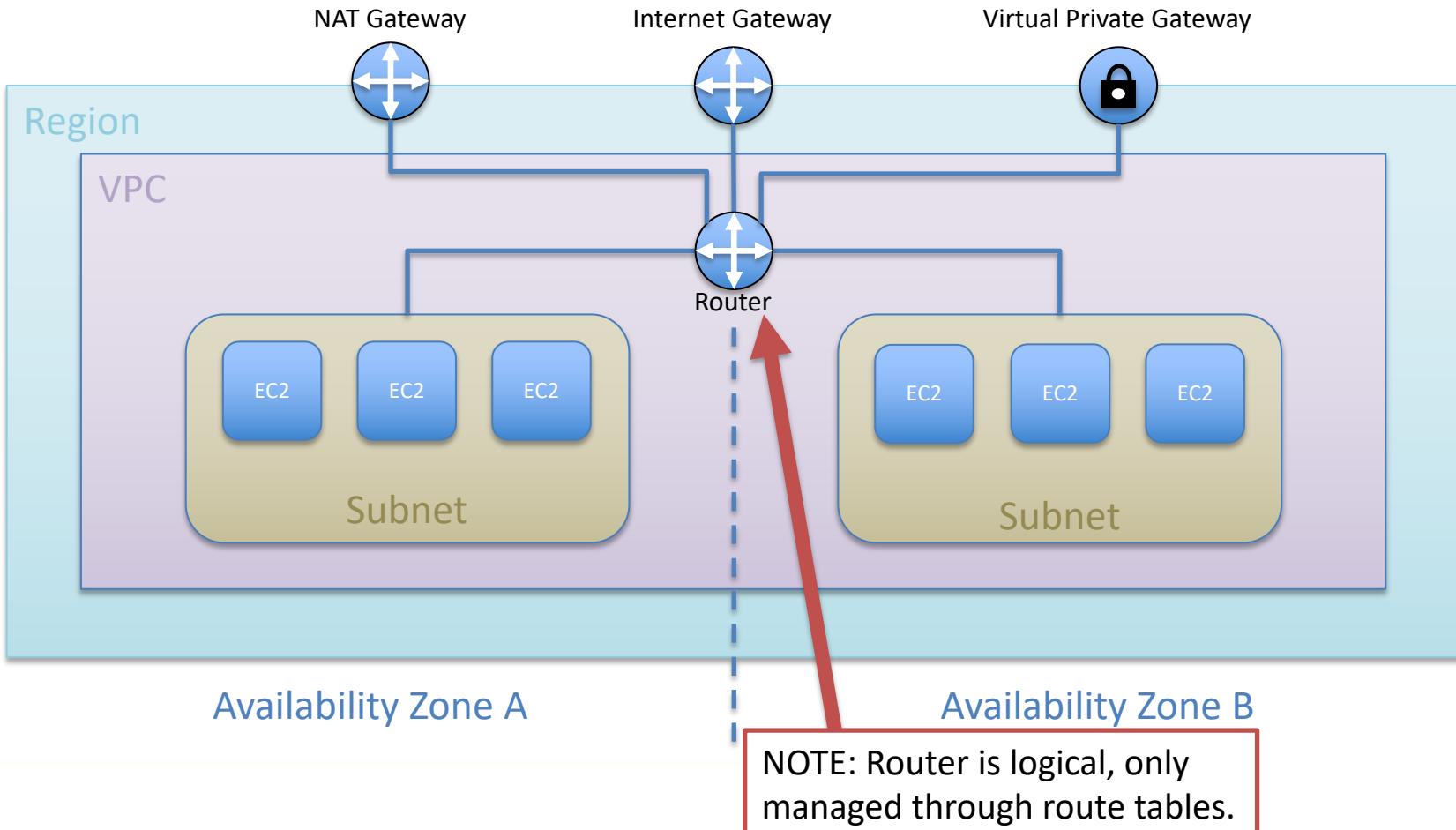
VPCs are in here.

Region

Availability Zones



VPC Components



VPC Networking & Security

Concepts/Terminology

Subnets

Route Tables

Security Groups

Network ACL

NAT
Gateway

Internet
Gateway
(IGW)

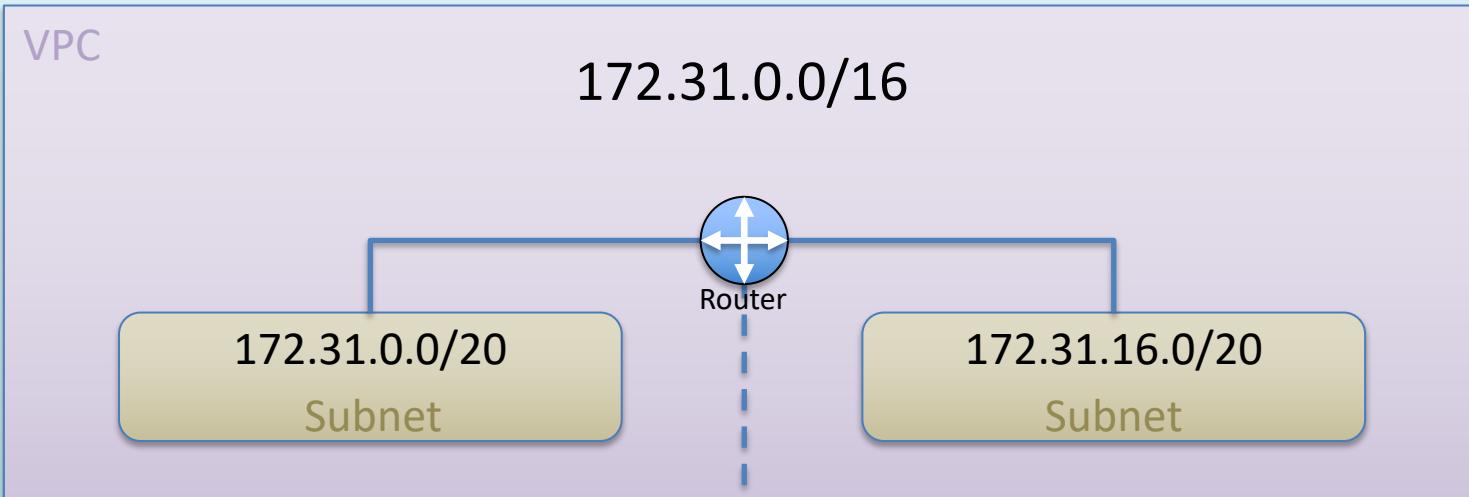
Egress-only
IGW

Elastic IP

VPC Networking & Security

Subnetting

Region

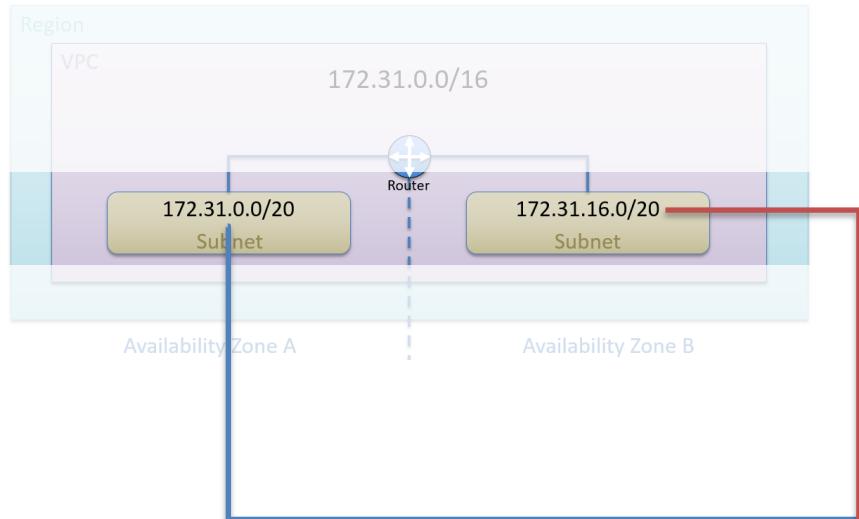


Availability Zone A

Availability Zone B

VPC Networking & Security

Subnets have Route Tables



Route Table ID: rtb-4b576222

Summary Routes Subnet Associations Route Propagation Tags

Edit View: All rules

Destination	Target	Status	Propagated
172.31.0.0/16	local	Active	No
0.0.0.0/0	igw-a2bed8cb	Active	No

rtb-4b576222

Summary Routes Subnet Associations

Edit Subnet IPv4 CIDR IPv6 CIDR

You do not have any subnet associations.
The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

Subnet	IPv4 CIDR	IPv6 CIDR
subnet-70af8219	172.31.0.0/20	-
subnet-ef99f694	172.31.16.0/20	-
subnet-641fa829	172.31.32.0/20	-

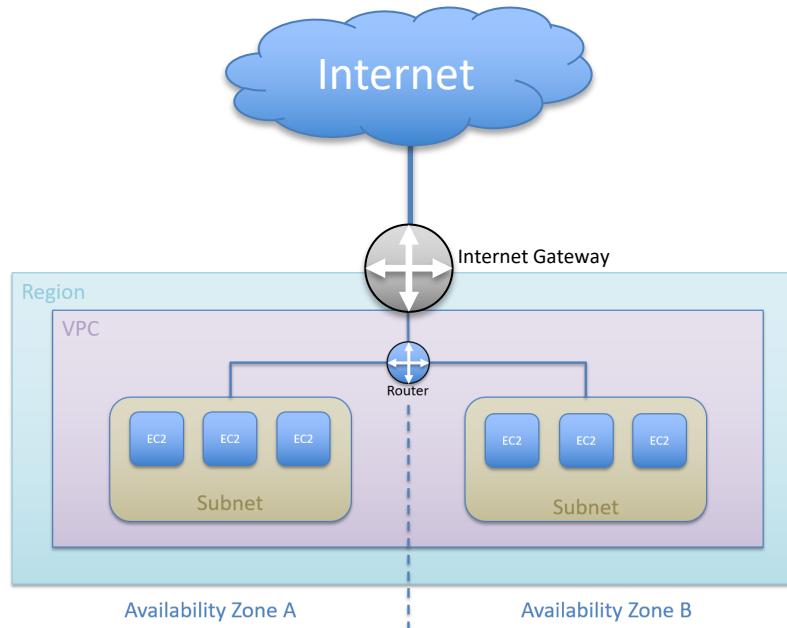
VPC Networking & Security

VPC Connectivity Options

- Internet
- Virtual Gateway IPSec
- VPC Peering
- Direct Connect Peering

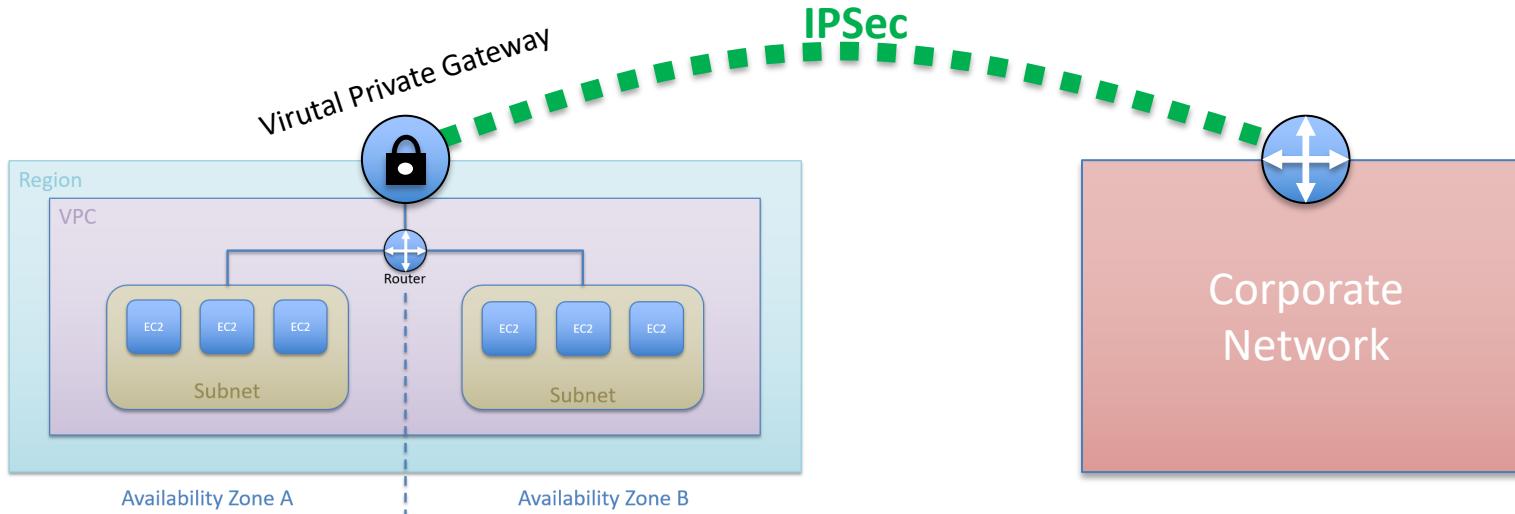
VPC Networking & Security

Internet



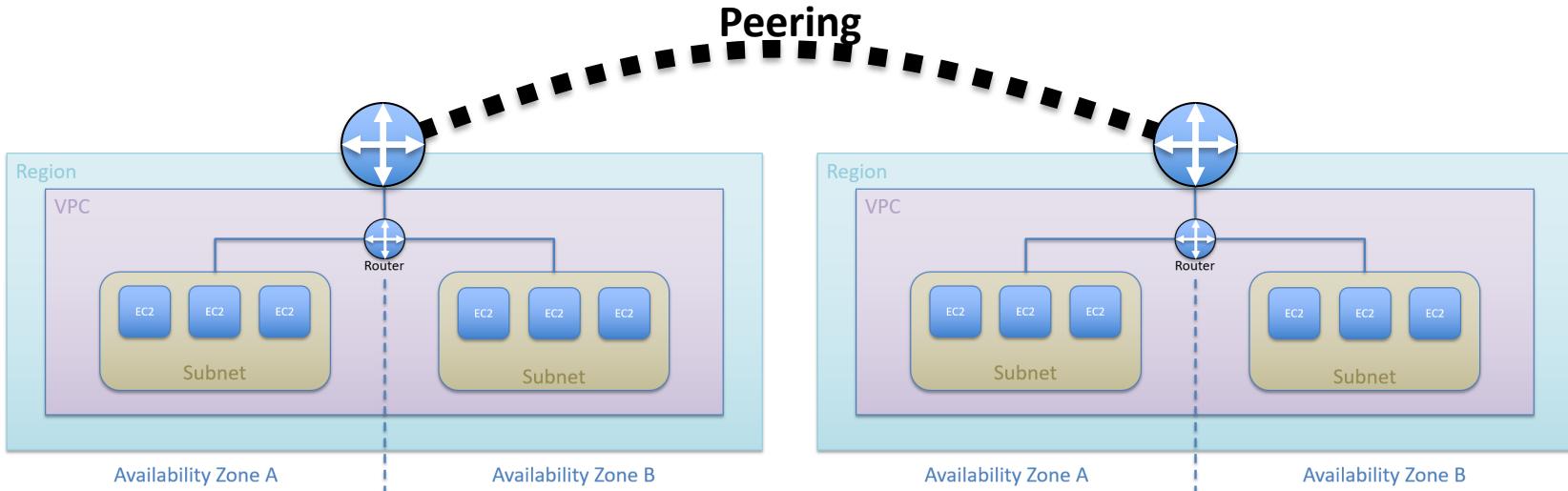
VPC Networking & Security

VPC Gateway IPSec



VPC Networking & Security

VPC Peering



VPC Networking & Security

Direct Connect

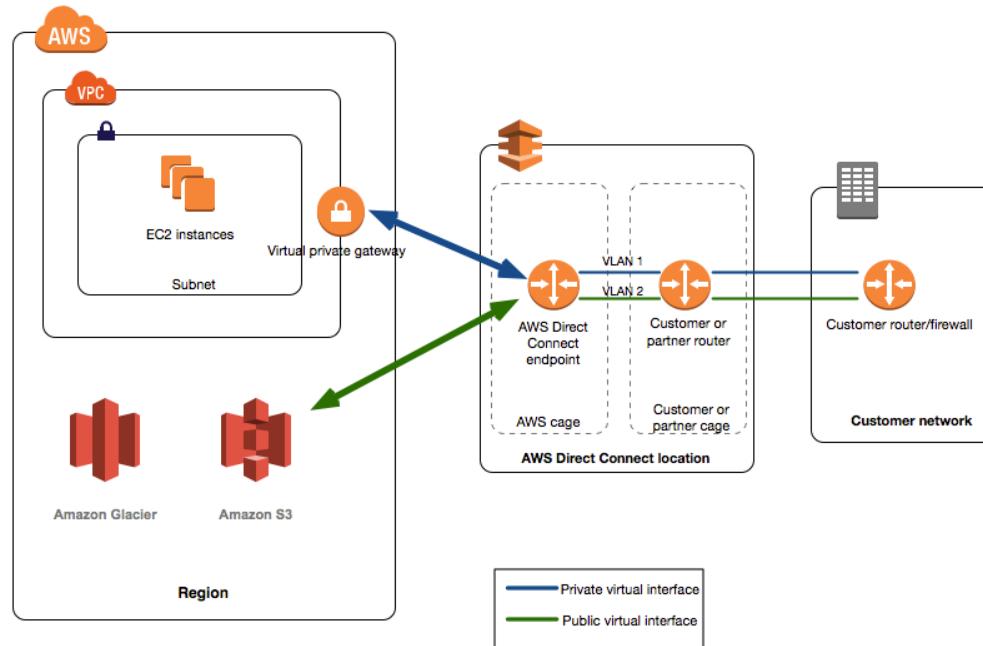
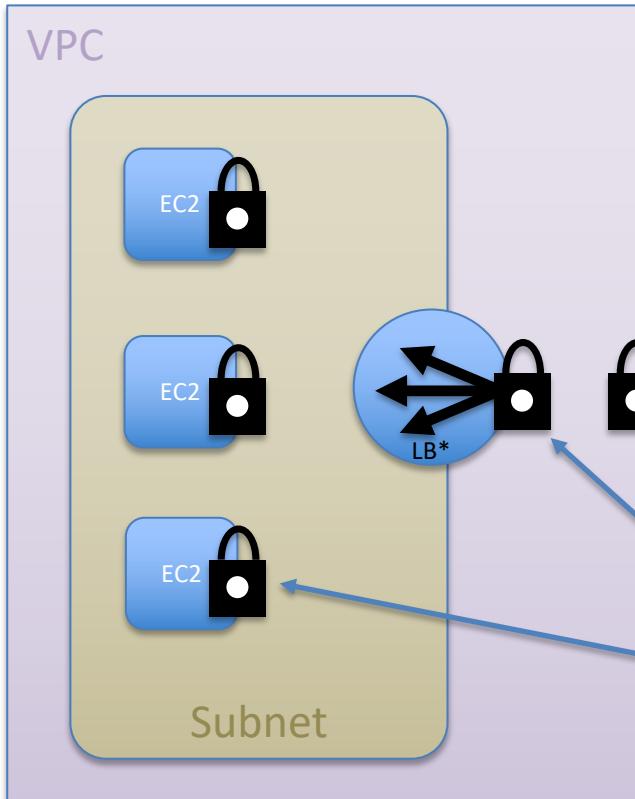


Image Source: Amazon

VPC Networking & Security



Security rules can be applied at several places in an AWS

- VPC
- Load Balancers*
- EC2 Instances

Network ACLs

Security Groups

* Classic Load Balancer

VPC Networking & Security

Subnet Network Access List

The screenshot shows the AWS VPC Dashboard with the "Network ACLs" section selected. It displays two Network ACLs: "acl-43c20b26" and "acl-7dc20b18". The "acl-43c20b26" details page is shown, featuring tabs for Summary, Inbound Rules, Outbound Rules, Subnet Associations, and Tags. The Inbound Rules table lists five rules:

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
1	ALL Traffic	ALL	ALL		DENY
98	ALL Traffic	ALL	ALL		DENY
99	ALL Traffic	ALL	ALL		DENY
100	ALL Traffic	ALL	ALL	0.0.0.0/0	ALLOW
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY

VPC Networking & Security

VPC Security Group

The screenshot shows the AWS VPC Management Console interface. The left sidebar navigation includes:

- VPC Dashboard
- Virtual Private Cloud
 - Your VPCs
 - Subnets
 - Route Tables
 - Internet Gateways
 - Egress Only Internet Gateways
 - DHCP Options Sets
 - Elastic IPs
 - Endpoints
 - NAT Gateways
 - Peering Connections
- Security
 - Network ACLs
 - Security Groups**
- VPN Connections
 - Customer Gateways
 - Virtual Private Gateways
 - VPN Connections

The main content area displays the "Security Group Actions" tab for "All security groups". A single entry is listed:

Name tag	Group ID	Group Name	VPC	Description
sg-a936f9c1	sg-a936f9c1	default	vpc-9bb9e4f2	default VPC security group

Below this, the details for the "sg-a936f9c1" security group are shown. The "Inbound Rules" tab is selected, displaying one rule:

Type	Protocol	Port Range	Source
ALL Traffic	ALL	ALL	sg-a936f9c1

At the bottom of the page, there are links for Feedback, English, and footer text: © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use.

VPC Networking & Security

VPC NAT Gateways

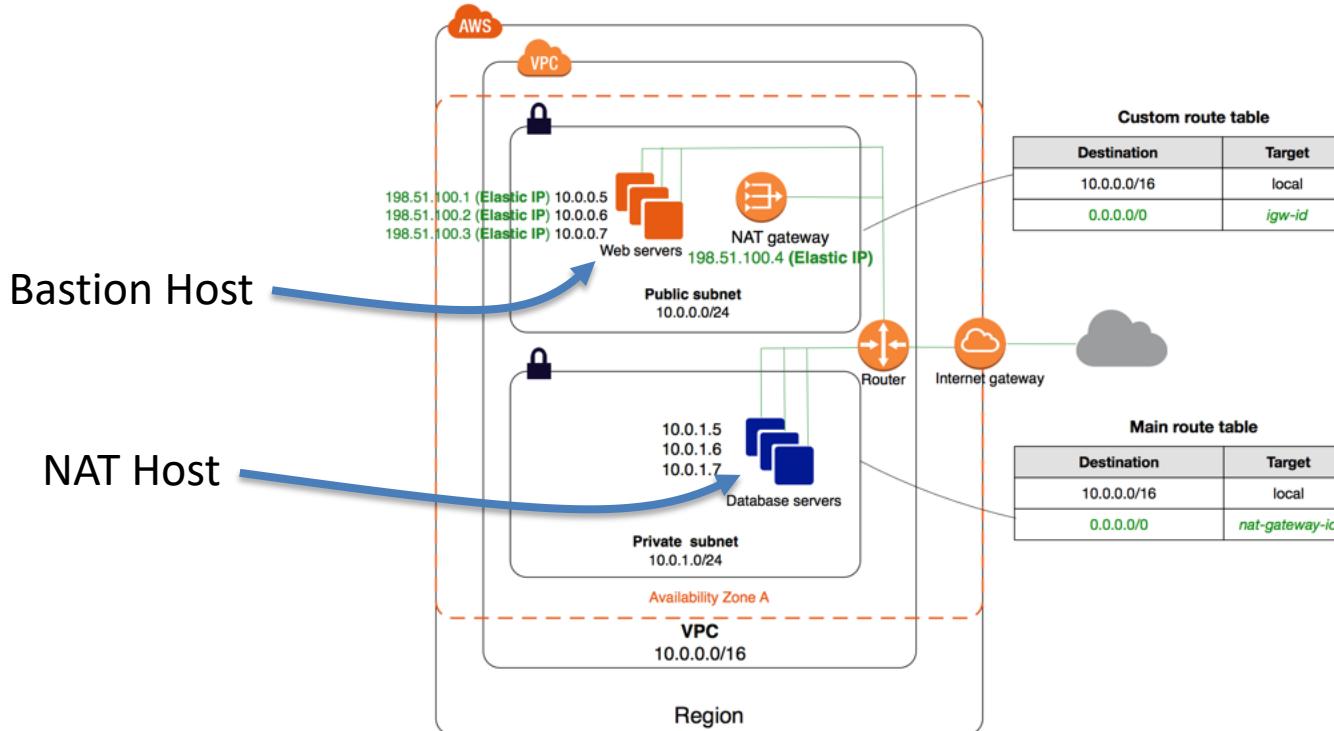


Image Source: Amazon

<https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-gateway.html>

VPC Flow Logs

“...capture information about the IP traffic going to and from network interfaces in your VPC.”

- Publish logs to:
 - Amazon CloudWatch Logs
 - Amazon S3
- Common Use Cases:
 - Troubleshoot traffic flow, security group issues.
 - Security monitoring

VPC Flow Logs

Flow Record Format:

```
<version> <account-id> <interface-id> <srcaddr>  
<dstaddr> <srcport> <dstport> <protocol> <packets>  
<bytes> <start> <end> <action> <log-status>
```

Permitted SSH

```
2 123456789010 eni-abc123de 172.31.16.139 172.31.16.21  
20641 22 6 20 4249 1418530010 1418530070 ACCEPT OK
```

Denied RDP

```
2 123456789010 eni-abc123de 172.31.9.69 172.31.9.12  
49761 3389 6 20 4249 1418530010 1418530070 REJECT OK
```

VPC Overview

VPC Walkthrough



AWS Certified Developer (Associate) Crash Course

Foundational Services
EC2

EC2 Overview

Elastic

Elastic Scalability

Compute

Servers
(Compute Instances)

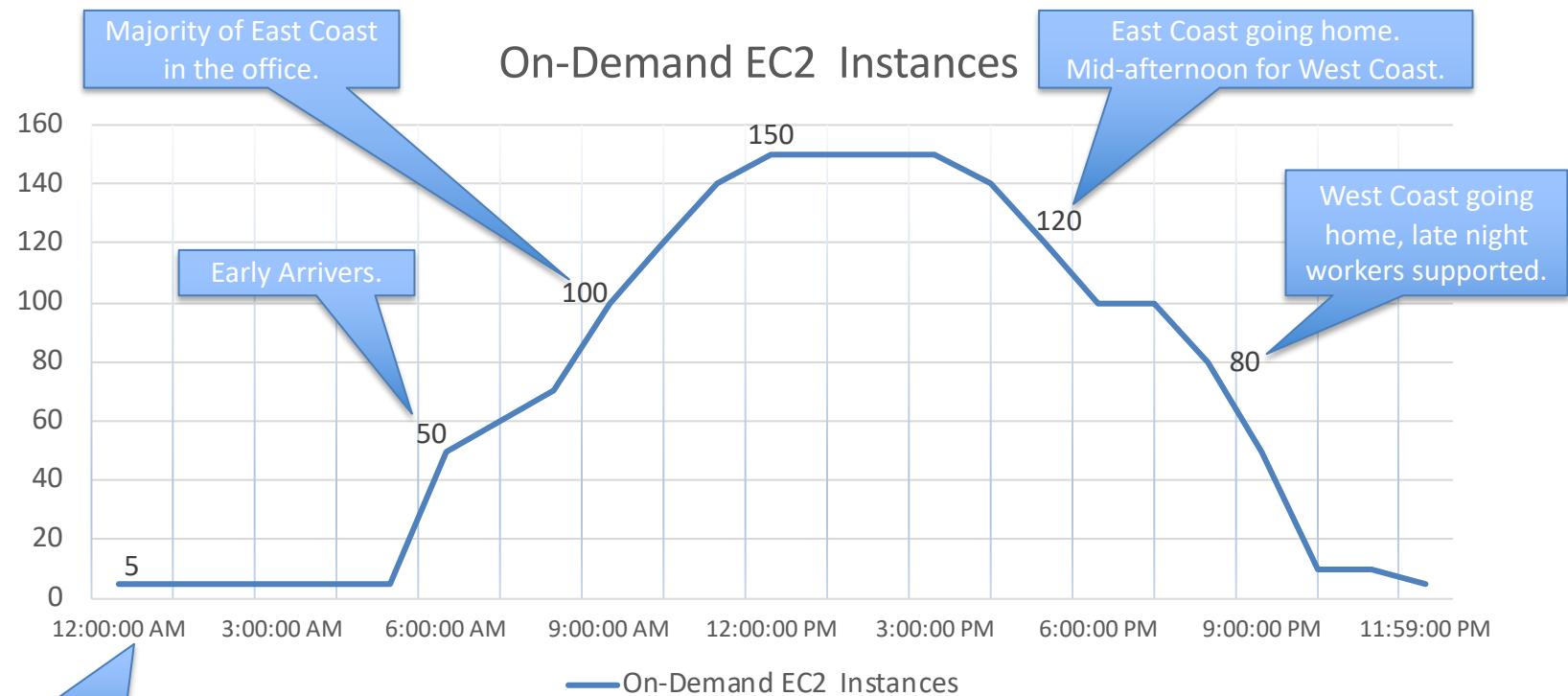
Cloud

In the cloud

- PaaS
- Scale capacity $\uparrow\downarrow$ on-demand
- Time-based scaling

- Previously based on Xen virtualization
- M5/C5 instances use Nitro Hypervisor
- Free Tier

EC2 Overview – Instance Scaling



EC2 Overview – Pricing

- No CapEx
- Four pricing options:

On-Demand

- Hourly
- No long-term commitment
- No upfront payments

EC2 Overview – Pricing

- No CapEx
- Four pricing options:

On-Demand

Spot Instances

- Hourly
 - No long-term commitment
 - No upfront payments
- Bid on spare compute capacity
 - Up to 90% discount

EC2 Overview – Pricing

- No CapEx
- Four pricing options:

On-Demand

Spot Instances

Reserved
Instances

- Hourly
- No long-term commitment
- No upfront payments

- Bid on spare compute capacity
- Up to 90% discount

- Reserves compute capacity for when needed
- Up to 75% discount

EC2 Overview – Pricing

- No CapEx
- Four pricing options:

On-Demand

- Hourly
- No long-term commitment
- No upfront payments

Spot Instances

- Bid on spare compute capacity
- Up to 90% discount

Reserved Instances

- Reserves compute capacity for when needed
- Up to 75% discount

Dedicated Hosts

- Physical EC2 server dedicated for your use
- Eases:
- Compliance
 - Server-bound Licenses

EC2 Overview – Pricing

- Pricing is Region-bound
 - Some regions are cheaper than others
- Example:
 - t2.micro
 - Linux
 - \$0.012/Hour



Linux	RHEL	SLES	Windows	Windows with SQL Standard	Windows with SQL Web
Windows with SQL Enterprise					
Region: US East (Ohio)					
vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage	
General Purpose - Current Generation					
t2.nano	1	Variable	0.5	EBS Only	\$0.0059 per Hour
t2.micro	1	Variable	1	EBS Only	\$0.012 per Hour
t2.small	1	Variable	2	EBS Only	\$0.023 per Hour
t2.medium	2	Variable	4	EBS Only	\$0.047 per Hour
t2.large	2	Variable	8	EBS Only	\$0.094 per Hour
t2.xlarge	4	Variable	16	EBS Only	\$0.188 per Hour
t2.2xlarge	8	Variable	32	EBS Only	\$0.376 per Hour
m4.large	2	6.5	8	EBS Only	\$0.1 per Hour
m4.xlarge	4	13	16	EBS Only	\$0.2 per Hour
m4.2xlarge	8	26	32	EBS Only	\$0.4 per Hour
m4.4xlarge	16	53.5	64	EBS Only	\$0.8 per Hour
m4.10xlarge	40	124.5	160	EBS Only	\$2 per Hour
m4.16xlarge	64	188	256	EBS Only	\$3.2 per Hour

<https://aws.amazon.com/ec2/pricing/on-demand/>

EC2 Overview – Instance Types

- EC2 instances are provisioned based on capabilities needed
- User selected
- Many to choose from
- Instance Type Categories:
 - General Purpose
 - GPU Graphics
 - Memory Optimized
 - Compute Optimized
 - GPU Compute
 - Storage Optimized

Step 2: Choose an Instance Type
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance types Current generation Show/Hide Columns

Currently selected: t2.xlarge (Variable ECUs, 4 vCPUs, 2.3 GHz, Intel Broadwell E5-2686v4, 16 GB memory, EBS only)

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
General purpose	t2.micro	1	1	EBS only	-	Low to Moderate	Yes
General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes
General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
General purpose	t3.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes
General purpose	t3.micro	2	1	EBS only	Yes	Up to 5 Gigabit	Yes
General purpose	t3.small	2	2	EBS only	Yes	Up to 5 Gigabit	Yes
General purpose	t3.medium	2	4	EBS only	Yes	Up to 5 Gigabit	Yes
General purpose	t3.large	2	8	EBS only	Yes	Up to 5 Gigabit	Yes
General purpose	t3.xlarge	4	16	EBS only	Yes	Up to 5 Gigabit	Yes
General purpose	t3.2xlarge	8	32	EBS only	Yes	Up to 5 Gigabit	Yes
General purpose	m5ad.large	2	8	1 x 75 (SSD)	Yes	Up to 10 Gigabit	Yes

138 Instance Types as of April 2019
256 Instance Types as of July 2020

EC2 Overview – Instance Types

General Purpose	Balance of compute, memory, and network resources. Use: Small/medium databases, data processing tasks that require additional memory.
Compute Optimized	Higher ratio of vCPUs to memory. Lowest cost per vCPU. High-end CPUs Use: High-traffic front end fleets, on-demand batch processing, distributed analytics, web servers, etc.
GPU Graphics	Provide GPUs along with high CPU performance. Large memory, high network speed. Use: Applications requiring high-performance graphics acceleration, such as 3D visualizations, 3D rendering, video encoding, and virtual reality.

EC2 Overview – Instance Types

GPU Compute	Provide general-purpose GPUs and high CPU performance. Large memory / high network speed Use: Applications requiring massive floating-point processing power, such as machine learning, high performance databases, computational fluid dynamics, etc.
Memory Optimized	Lowest cost per GB of RAM. High-end memory Use: Database applications, memcached and other distributed caches, larger deployments of enterprise applications like SAP and Microsoft SharePoint.
Storage Optimized	Provide you with direct-attached storage options optimized for applications with specific disk I/O and storage capacity requirements. Use: Databases which benefit from very high random I/O performance, and low request latency of direct-attached SSDs.

EC2 Overview – O/S's

Linux

- Amazon Linux
- Amazon Linux 2
- SUSE Linux Enterprise Server
- Red Hat Enterprise Linux
- Ubuntu Server

Windows

- Server 2019
 - Server 2016
 - Server 2012 R2
 - Server 2012
 - Server 2008
-
- Sub-options for:
 - SQL Server Base
 - SQL Server Web
 - SQL Server Standard

EC2 Overview – O/S's

Linux

- Amazon Linux
- SUSE Linux Enterprise Server
- Red Hat Enterprise Linux
- Ubuntu Server

Windows

- Server 2019
- Server 2016
- Server 2012 R2
- Server 2012
- Server 2008

(Base images only)

Free Tier Eligible

EC2 Overview – Storage

EC2 Instance Options for Root Device:

Instance Store

- Access storage from disks that are physically attached to the host computer
- Temporary block-level storage for instances
- Data persists only during the life of the instance
- If you stop or terminate an instance, any data on instance store volumes is lost

EC2 Overview – Storage

EC2 Instance Options for Root Device:

Instance Store

- Access storage from disks that are physically attached to the host computer
 - Temporary block-level storage for instances
 - Data persists only during the life of the instance
 - If you stop or terminate an instance, any data on instance store volumes is lost
-

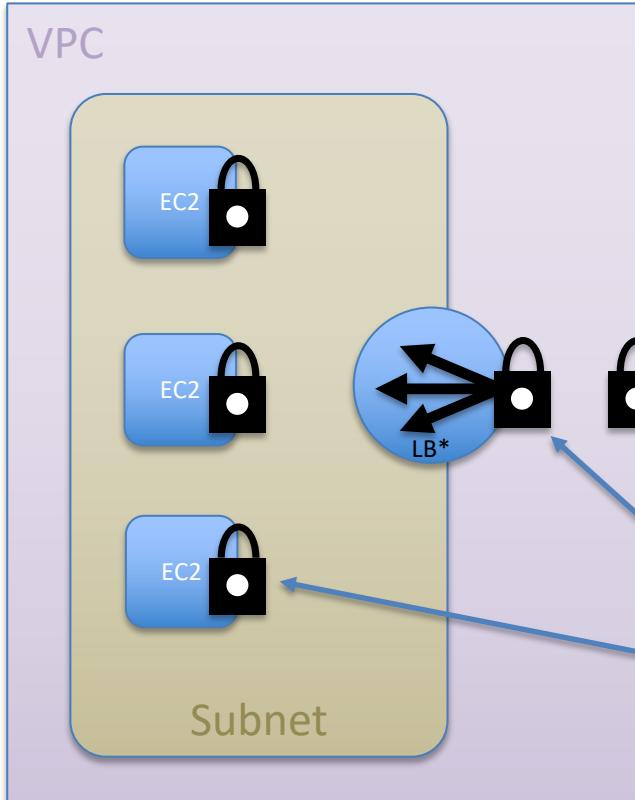
EBS Volume

- Elastic Block Storage
- Block-level storage volumes that you can attach to a running instance
- Use Amazon EBS as a primary storage device for data that requires frequent and granular updates. Example: Database files
- Behaves like a raw, unformatted, external block device that you can attach to a single instance
- Volume persists independently from the running life of an instance
- Can detach/attach to instances at will
- Non-Volatile Memory Express (NVMe) available on new generation instances. Provides 3.3 million random IOPS.

EC2 Overview – Networking

- Instances are assigned a Private RFC1918 address when provisioned.
 - Can assign public/routable IP address.
 - Standard network conventions apply.
- 
- Subnet-to-subnet communication needs a “router”, route tables.
 - Hosts in same subnet can talk to each other provided communication is permitted. Default is deny.
 - Further details on VPC Network is in the VPC section of this course.

EC2 Overview – Security



Security controls can be applied at several places in an AWS

- VPC
- Load Balancers*
- EC2 Instances

Network ACLs

Security Groups

* Classic Load Balancer

EC2 Overview – Security

EC2 Instance Security Group

The screenshot shows the AWS EC2 Dashboard with the 'Security Groups' section selected. The main pane displays a list of security groups, with 'sg-df51ceba' selected. The 'Inbound' tab is active, showing the following rules:

Type	Protocol	Port Range	Source
HTTP	TCP	80	0.0.0.0/0
SSH	TCP	22	0.0.0.0/0
Custom TCP Rule	TCP	8079 - 8080	0.0.0.0/0
Custom TCP Rule	TCP	3001	0.0.0.0/0
Custom TCP Rule	TCP	3000	0.0.0.0/0
All ICMP	All	N/A	0.0.0.0/0

EC2 Overview

EC2 Instance Metadata

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/  
ami-id  
ami-launch-index  
ami-manifest-path  
block-device-mapping/  
events/  
hostname  
iam/  
instance-action  
instance-id  
instance-type  
local-hostname  
local-ipv4  
mac  
metrics/  
network/  
placement/  
profile  
public-hostname  
public-ipv4  
public-keys/  
reservation-id  
security-groups  
services/
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-hostname  
ec2-203-0-113-25.compute-1.amazonaws.com
```

EC2 Overview

EC2 Demo



AWS Certified Developer (Associate) Crash Course

Foundational Services

Route53

Route53 Overview

- AWS Managed Domain Name System (DNS)
- Provides translation of names to IP addresses
 - e.g. www.example.com -> 192.2.3.10
- Stores name to IP mappings.
- Answers resolution queries from clients
- IPv6 Compliant
- Changes propagated worldwide within 60 seconds. Record TTL still applies.

Route53 Overview – Features

- **Traffic Management**
Direct users to the best endpoint based on:
geoproximity, latency, health, etc.
- **Latency Based Routing**
Route end users to the AWS region providing
the lowest latency.
- **Geo DNS**
Route end users based on their geographic
location.

Route53 Overview – Features

- **Private DNS for VPC**
Manage custom domain names for your internal AWS resources without exposing DNS data to the public Internet.
- **DNS Failover**
Automatically route your website visitors to an alternate location to avoid site outages.
- **Health Checks and Monitoring**
Monitor the health and performance of your application as well as your web servers and other resources.

Route53 Overview – Features

- **Domain Registration**

Search for, purchase, and manage domain names through Route53. Available through CLI/SDK as well.

- **CloudFront & S3 Zone Apex Support**

Access CloudFront and S3 through root domain, e.g. example.com instead of www.example.com.

- **ELB Integration**

Elastic Load Balancing (ELB) for dynamic IP alias assignment.

Route53 Overview – Features

- **Weighted Round Robin**

Specify the frequency (“weights”) with which different DNS responses are returned to end users.
E.g. A/B testing, region balancing.

- **Management**

Managed through web console and/or AWS CLI/API.
IAM permissions

Route53 Overview – Features

Record Type Support

Type	Description
A	Address Record
AAAA	IPv6 Address Record
CNAME	Canonical Name
CAA	Certification Authority Authorization
MX	Mail Exchange Record
NAPTR	Name Authority Pointer Record

Route53 Overview – Features

Record Type Support

Type	Description
NS	Name Server Record
PTR	Pointer Record
SOA	Start of Authority
SPF	Sender Policy Framework
TXT	Text Record

Route53 Routing Policies

Routing Policy

Simple Routing

No unique routing, e.g. weighted or latency.

Failover Routing

```
if (resource.healthy) route_to_it()  
else route_elsewhere()
```

Geolocation Routing

Choose the server based on the geographic location of the user(s).

E.g. Send EU users to Frankfurt.

Geoproximity Routing

Route traffic to your resources based on the geographic location of your users and your resources. Requires Traffic Flow, optional bias rules.

Route53 Routing Policies

Routing Policy

Latency-based Routing

If multi-region, route to region with lowest latency.

Multivalue Answer Routing

Return multiple answers for a query. Route53 will only include healthy hosts.

Weighted Routing

Associate multiple resources and route to them based on a weight. E.g. 60%/40%

Route53 Overview

Route53 Demo



AWS Certified Developer
(Associate) Crash Course
Database and
Storage Services

A large, semi-transparent play button icon is positioned on the left side of the slide, consisting of a white triangle pointing right inside a white circle with a gray outline.

AWS Certified Developer (Associate) Crash Course

Simple Storage Service

S3 Overview

Simple

Pretty easy to use

Storage

Cloud Storage

Service

AWS Service

- Intentionally minimal
- REST and SOAP Interfaces
- Region-based pricing
- Fine-grained permissions

- Store an infinite amount of data
- Objects up to 5TB in size
- Created with Console, CLI, API

S3 Overview

S3 Management Console  Secure | https://s3.console.aws.amazon.com/s3/home?region=us-east-2# Nick Garner Global Support Documentation

Services Resource Groups

Identify optimal storage classes with S3 Analytics - Storage Class Analysis. [Learn More »](#)

Amazon S3

+ Create bucket Delete bucket Empty bucket

- Buckets - Regions

Discover the new console Quick tips

You do not have any buckets. Here is how to get started with Amazon S3.

Create a new bucket



Buckets are globally unique containers for everything that you store in Amazon S3.

Upload your data



After you create a bucket, you can upload your objects (for example, your photo or video files).

Set up your permissions



By default, the permissions on an object are private, but you can set up access control policies to grant permissions to others.

Learn more Learn more Learn more

Get started

176.32.114.59 English (US) © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

S3 Overview

Data Consistency

- Read-after-write consistency for new objects.
- Eventually consistent
 - Read after write might return old data.
 - List after write might not have new object.
- New object PUT: read-after-write
- Overwrite PUTS and DELETES: eventually consistent

S3 Overview

Storage Classes

- Three storage classes for objects.
 - General Purpose (1 class option)
 - Infrequent Access (2 class options)

Class Options

- Amazon S3 Standard
- Amazon S3 Standard-Infrequent Access
- Amazon S3 One Zone-Infrequent Access

S3 Overview

Amazon S3 Standard

High durability, availability, and performance. Data stored in three AZs.

Key Features:

- Low latency and high throughput performance
- Durability of 99.99999999% of objects
- Can survive single AZ destruction
- 99.99% availability/year
- Backed with the Amazon S3 Service Level Agreement
 - <https://aws.amazon.com/s3/sla/>
- Supports SSL for data in transit and encryption of data at rest
- Lifecycle management for automatic migration of objects

S3 Overview

Amazon S3 Standard-Infrequent Access

For: data that is accessed less frequently, but requires rapid access. Data stored in three AZs.

Key Features:

- Same latency and throughput performance of S3 Standard
- Designed for durability of 99.99999999% of objects
- Can survive single AZ destruction
- Designed for **99.9%** availability over a given year
- Backed with the Amazon S3 Service Level Agreement
 - <https://aws.amazon.com/s3/sla/>
- Supports SSL for data in transit and encryption of data at rest
- Lifecycle management for automatic migration of objects

S3 Overview

Amazon S3 One Zone-Infrequent Access

For: data that is accessed less frequently, but requires rapid access. Data only stored in one AZ.

Key Features:

- Same latency and throughput performance of S3 Standard
- Designed for durability of 99.99999999% of objects
- Designed for **99.5%** availability over a given year
- Backed with the Amazon S3 Service Level Agreement
 - <https://aws.amazon.com/s3/sla/>
- Supports SSL for data in transit and encryption of data at rest
- Lifecycle management for automatic migration of objects

S3 Overview

Reduced Redundancy Storage (RRS)

- User can elect RRS.
- For storing non-critical, reproducible data.
- 99.99% durability of objects over a given year.
- Average expected loss of 0.01% of objects annually.

NOTE: RRS is covered here for completeness. RRS is legacy and no longer recommended and is more expensive than S3 standard pricing. RRS has been replaced by standard and one-zone infrequent access.

S3 Overview

S3 Region Availability (as of July 2020)

US East (N. Virginia)	Asia Pacific (Tokyo, Osaka/Local)
US East (Ohio)	China (Beijing)
US West (N. California)	China (Ningxia)
US West (Oregon)	South America (São Paulo)
Canada (Central)	EU (Frankfurt)
Asia Pacific (Mumbai)	EU (Ireland)
Asia Pacific (Seoul)	EU (London)
Asia Pacific (Singapore)	EU (Paris)
Asia Pacific (Sydney)	EU (Stockholm)
Asia Pacific (Hong Kong)	EU (Milan)
Africa (Capetown)	Middle East (Bahrain)

S3 Components

Buckets

Objects

Keys

S3 Overview

Buckets

- Container for objects stored in Amazon S3
- Every object is stored in a bucket
- <http://awsdevguru.s3.amazonaws.com/images/rainbow.jpg>

S3 Overview

Buckets

- Container for objects stored in Amazon S3
- Every object is stored in a bucket
- <http://awsdevguru.s3.amazonaws.com/images/rainbow.jpg>
Bucket Object

Bucket purposes:

- Organize the Amazon S3 namespace
- Identify the account responsible
- Aggregation for usage reporting
- Deployed in a specific region

Versioning:

- Buckets can be configured so objects are versioned

S3 Overview

Objects

- Entities stored in Amazon S3
- Consist of:
 - Object Data: Opaque to Amazon S3
 - Metadata: Name-value pairs that describe the object
 - Default metadata:
 - Date last modified
 - standard HTTP metadata, such as Content-Type
 - Custom Metadata:
 - Specified when the object is stored
- Objects can be encrypted
 - Client-side and server-side options

S3 Overview

Keys

- Unique identifier for an object.
- Every object has one key.
- Bucket + Key + Version identifies all objects.
- Every object in Amazon S3 can be addressed with combination of:
 - The web service endpoint
 - Bucket name
 - Key
 - Version (optional)

S3 Overview

Empty Bucket

The screenshot shows the Amazon S3 Management Console interface. At the top, the URL is https://s3.console.aws.amazon.com/s3/buckets/awsdevguru/?region=us-east-2&tab=overview. The navigation bar includes Services, Resource Groups, Nick Garner, Global, and Support. Below the navigation, the breadcrumb path is Amazon S3 > awsdevguru. The tabs at the top are Overview (selected), Properties, Permissions, and Management. Below the tabs are buttons for Upload, Create folder, and More. The region is set to US East (Ohio). The main content area displays a message: "This bucket is empty. Upload new objects to get started." It features three sections: "Upload an object" with an icon of a bucket and a paperclip, "Set object properties" with an icon of two people and a plus sign, and "Set object permissions" with an icon of a database and gears. Each section has a "Learn more" link and a "Get started" button.

This bucket is empty. Upload new objects to get started.

Upload an object

Buckets are globally unique containers for everything that you store in Amazon S3.

Learn more

Set object properties

After you create a bucket, you can upload your objects (for example, your photo or video files).

Learn more

Get started

Set object permissions

By default, the permissions on an object are private, but you can set up access control policies to grant permissions to others.

Learn more

176.32.114.59

S3 Overview

File Added

Object Path

Object

Storage Class

The screenshot shows the AWS S3 Management Console interface. At the top, there's a navigation bar with the AWS logo, 'Services', 'Resource Groups', and user information ('Nick Garner', 'Global', 'Support'). Below the navigation bar, the 'Overview' tab is selected in a row of tabs labeled 'Overview', 'Properties', 'Permissions', and 'Management'. A search bar is present above the main content area. In the center, there's a table listing three objects:

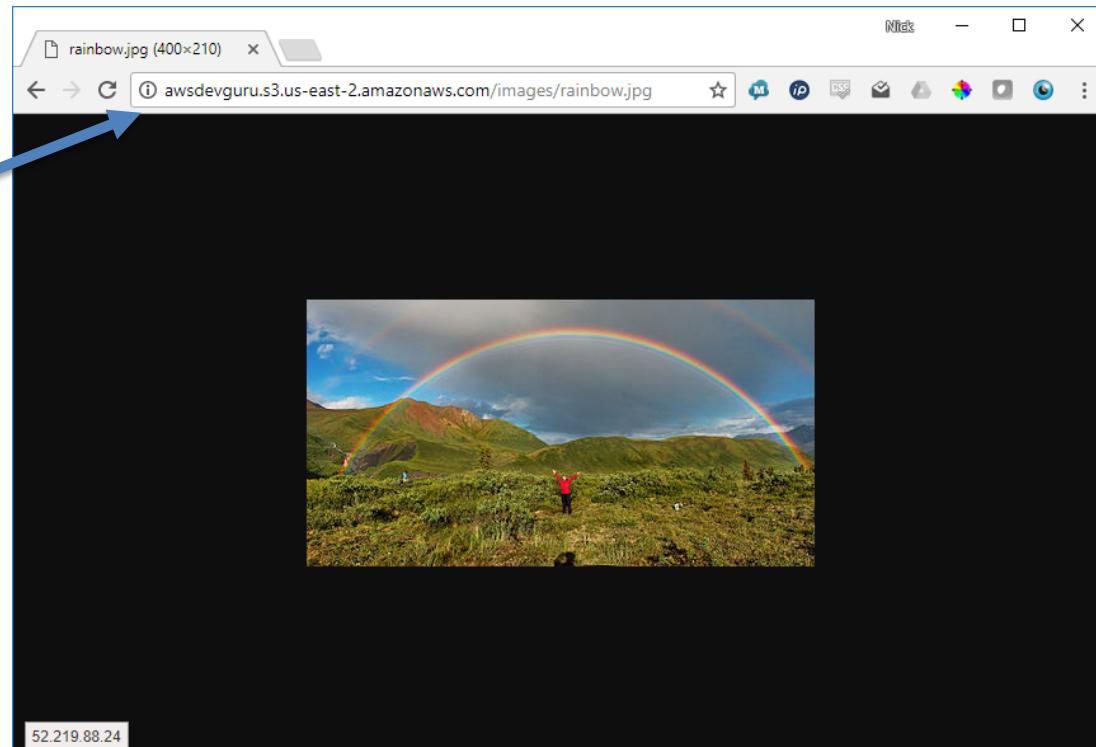
Name	Last modified	Size	Storage class
rainbow.png	Aug 10, 2018 11:05:09 AM GMT-0700	9.5 KB	Standard
double_rainbow.png	Aug 10, 2018 11:20:29 AM GMT-0700	56.3 KB	Standard-IA
pot_of_gold.png	Aug 10, 2018 11:20:30 AM GMT-0700	94.6 KB	One Zone-IA

At the bottom of the table, it says 'Viewing 1 to 3'. The footer of the console includes 'Operations' (0 In progress, 4 Success, 0 Error) and a status bar with the IP address '54.239.31.83'.

S3 Overview

Made Public

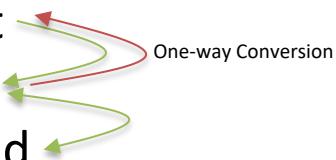
URL Access



S3 Overview

Version Control

- **Versioning:** keeping multiple variants of an object in the same bucket. Allows for recovery of data from user action and/or app failure.
- Object deletion in a version-enabled bucket does not delete the object, sets Delete Marker. Standard GET -> 404.
- Can perma-delete with “DELETE Object versionId”
- Bucket States:
 - Unversioned, default
 - Versioning-enabled
 - Versioning-suspended



Lifecycle Management

- Manage cost through lifecycle management of objects.
- **Lifecycle Configuration:** set of rules that define actions that S3 applies to a group of objects.
- Two actions:
 - **Transition actions**—When to transition an object to a different storage class. E.g. transition to STANDARD_IA after 30 days, to Glacier after 1 year.
 - **Expiration actions**—Define when objects expire, automatically deleted.

Transfer Acceleration

- Data ingestion via Edge Locations
- Transferred from edge to bucket through Amazon's network.
- Access acceleration-enabled bucket:
bucketname.s3-accelerate.amazonaws.com

Speedtest:

<http://s3-accelerate-speedtest.s3-accelerate.amazonaws.com/en/accelerate-speed-comparsion.html>

S3 Overview

Transfer Acceleration

Speedtest:

<http://s3-accelerate-speedtest.s3-accelerate.amazonaws.com/en/accelerate-speed-comparsion.html>

From Sunnyvale, CA:

Singapore (AP-SOUTHEAST-1)	6% faster	Sydney (AP-SOUTHEAST-2)	10% faster	São Paulo (SA-EAST-1)	7% faster
S3 Direct Upload Speed		S3 Direct Upload Speed		S3 Direct Upload Speed	
Upload complete		Upload complete		Upload complete	
S3 Accelerated Transfer Upload Speed		S3 Accelerated Transfer Upload Speed		S3 Accelerated Transfer Upload Speed	
Upload complete		Upload complete		Upload complete	
Mumbai (AP-SOUTH-1)	34% faster	Ohio (US-EAST-2)	2% faster	Canada Central (CA-CENTRAL-1)	5% faster
S3 Direct Upload Speed		S3 Direct Upload Speed		S3 Direct Upload Speed	
Upload complete		Upload complete		Upload complete	
S3 Accelerated Transfer Upload Speed		S3 Accelerated Transfer Upload Speed		S3 Accelerated Transfer Upload Speed	
Upload complete		Upload complete		Upload complete	

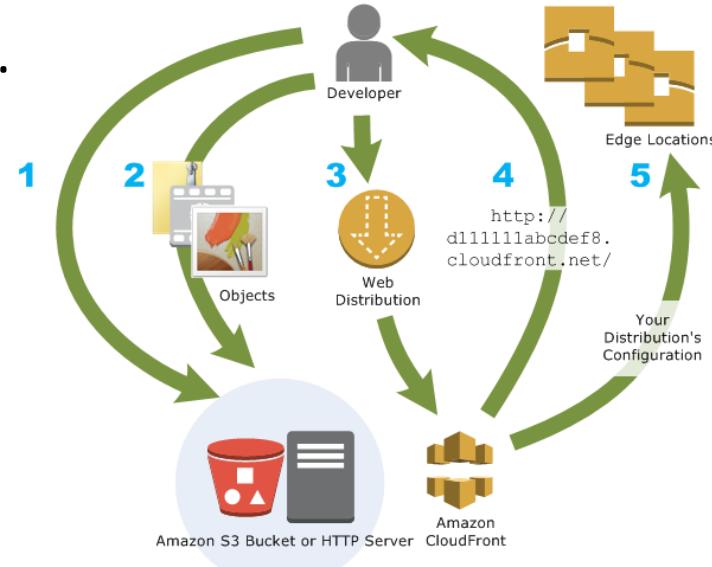
This speed checker uses multipart uploads to transfer a file from your browser to various Amazon S3 regions with and without Amazon S3 Transfer Acceleration. It compares the speed results and shows the percentage difference for every region.

Note: In general, the farther away you are from an Amazon S3 region, the higher the speed improvement you can expect from using Amazon S3 Transfer Acceleration. If you see similar speed results with and without the acceleration, your upload bandwidth or a system constraint might be limiting your speed.

Content Delivery Network

Cloudfront

1. Specify **origin servers**, source of your data.
 - S3 – or – own HTTP server
 - Original version of the files.
 - Custom Origin – non-AWS HTTP server
 - Adobe Media Server RTMP? Must use S3.
 - If origin == S3
 - objects in bucket publicly readable, or
 - Private, controlled access, Signed URLs/Cookies



Content Delivery Network

Cloudfront

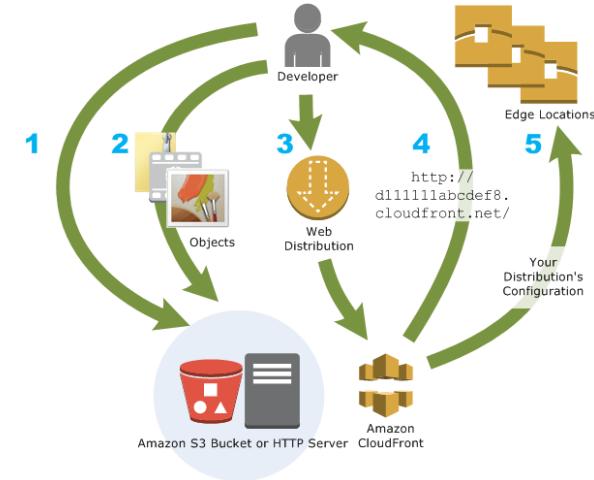
3. Create a CloudFront **distribution**

- Specifies the origin
- Log all requests?
- Enable now?

4. Assign Domain Name

5. Distribution configuration sent to edge locations

- Content not sent to edge locations.
- Cached on first view request.



S3 Best Practices

- Use IAM Permissions, no Full Control.
- Enable Bucket Encryption for data at rest.
 - Use S3 SSE-KMS for key management.
 - Use customer provided master key.
- Use SSL to secure data in transit.
- Enable Access Logging for S3 buckets.
- Use CORS to restrict access to objects from public.
- Use DNS compliant bucket names
 - A-Z, a-z, 0-9, -

S3 Best Practices

- Enable Versioning
 - Protect from overwrites and deletions
 - Retrieve/Restore deleted objects or rollback
- Enable MFA delete
- Backup Buckets

Versioning does not protect against bucket deletion
- Use Cross Region replication
- Use VPC S3 endpoints when communicating from VPC

S3 Overview

S3 Demo

A large, light-gray circular icon containing a white play triangle, positioned on the left side of the slide.

AWS Certified Developer (Associate) Crash Course

Relational Database Service

RDS Overview

Available RDS Database Engines



Engine	Edition
Amazon Aurora	MySQL & PostgreSQL Compliant
MySQL	Standard Community Edition
MariaDB	MariaDb Community Edition
PostgreSQL	Standard PostgreSQL
Oracle	Oracle EE, SE, SE One, SE Two
Microsoft SQL Server	Express, Web, Server SE, Server EE

RDS Overview

Amazon Aurora

- MySQL- and PostgreSQL-compatible
- <\$1/day
- 5 times the throughput of MySQL
- 3 times the throughput of PostgreSQL
- Up to 64TB of auto-scaling SSD storage
- 6-way replication across three Availability Zones
- Up to 15 Read Replicas with sub-10ms replica lag
- Automatic monitoring and failover in less than 30 seconds
- Aurora Serverless in preview (Aug. 2018)

RDS Overview

MySQL

- Open source community edition
- Database size up to 6 TB
- Up to 32 vCPUs and 244 GiB Memory
- Automated backup
- Point-in-time recovery
- Supports cross-region read replicas
- **Free tier eligible**

RDS Overview

PostgreSQL

- Open-source object-relational database
- Supports multiple extensions that add even more functionality to the database
- “The most Oracle-compatible open-source database.”
- **Free tier eligible**

RDS Overview

MariaDB

- MySQL-compatible database
- Database size up to 6 TB
- Up to 32 vCPUs and 244 GiB Memory
- Automated backup
- Point-in-time recovery
- Supports cross-region read replicas
- Supports global transaction ID (GTID) and thread pooling
- **Free tier eligible**

RDS Overview

Oracle

- Oracle Enterprise Edition
- Oracle Standard Edition
 - Up to 32 vCPUs
- Oracle Standard Edition One
 - Up to 16 vCPUs
- Oracle Standard Edition Two
 - Up to 16 vCPUs

RDS Overview

Microsoft SQL Server

- SQL Server Express
 - **Free tier eligible**
 - Database size up to 10GB
- SQL Server Web
 - Licensed to support public and Internet-accessible webpages, websites, web applications, and web services
- SQL Server Standard Edition
- SQL Server Enterprise Edition

RDS Overview

RDS Interfaces

- RDS Console
- AWS CLI
- Programmatic Interfaces
 - AWS SDKs
 - Libraries
 - RDS API

RDS Overview

RDS Pricing

Based on usage of components.

Instance Class	E.g., micro, small, large, xlarge
Running Time	Based on instance-hour
Storage	Per-GB per-month
I/O Requests/Month	Total number of storage I/O requests that you have made in a billing cycle
Backup Storage	Storage used for automated backups and snapshots
Data Transfer	Internet data transfer in and out of your DB instance

<https://aws.amazon.com/rds/pricing>

RDS Configuration

- Each engine has specific configuration options
- Master username/password specified during launch
- Production Option
 - Multi-AZ deployment
 - Provisioned IOPS
- Dev/Test Option
 - Not for production.
 - Free tier eligible
- Allocated storage defined during launch.
- Can be scaled without downtime.

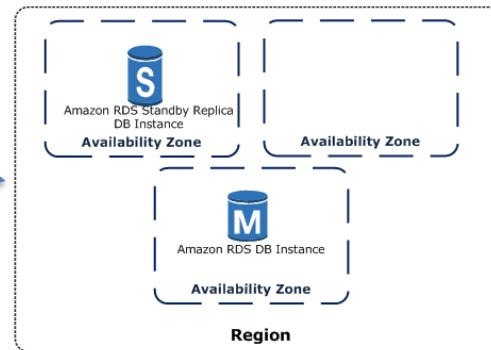


Image Source: Amazon

RDS Configuration

The screenshot shows the 'Specify DB Details' step of the AWS RDS MySQL configuration wizard. The left sidebar lists steps: Step 1: Select Engine, Step 2: Production?, Step 3: Specify DB Details (highlighted in blue), and Step 4: Configure Advanced Settings.

Free Tier
The Amazon RDS Free Tier provides a single db.t2.micro instance as well as up to 20 GB of storage, allowing new AWS customers to gain hands-on experience with Amazon RDS. Learn more about the RDS Free Tier and the instance restrictions [here](#).

Only show options that are eligible for RDS Free Tier

Instance Specifications

DB Engine: mysql
License Model: general-public-license
DB Engine Version: MySQL 5.6.35

Review the [Known Issues/Limitations](#) to learn about potential compatibility issues with specific database versions.

Billing estimate is based on on-demand usage as described in [Amazon RDS Pricing](#). Estimate does not include costs for backup storage, IOs (if applicable), or data transfer.

The following selections disqualify the instance from being eligible for the free tier:

- Multi-AZ Deployment
- DB Instance Class

You can receive a significant savings over on-demand instance costs with [Reserved Instances](#).

Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#).

Settings

DB Instance Identifier*: testdb
Master Username*: root
Master Password*:
Confirm Password*:

Retype the value you specified for Master Password.

52.95.18.74 back English (US) © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

RDS MySQL Configuration Options

RDS Configuration

Network & Security



VPC* Default VPC (vpc-9bb9e4f2)

Subnet Group default

Publicly Accessible Yes

Availability Zone No Preference

VPC Security Group(s) Create new Security Group

default (VPC)

Database Options

Database Name

Note: if no database name is specified then no initial MySQL database will be created on the DB Instance.

Database Port 3306

DB Parameter Group default:mysql5.6

Option Group default:mysql-5-6

Copy Tags To Snapshots

Enable IAM DB Authentication No Preference

Enable Encryption No

Backup

Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to detail [here](#).

Backup Retention Period 7 days

Backup Window No Preference

Monitoring

Enable Enhanced Monitoring No

Maintenance

Auto Minor Version Upgrade Yes

Maintenance Window No Preference

RDS Configuration

DB Parameter Groups

- Contains engine configuration values that can be applied to one or more DB instances of the same instance type
- Default DB parameter group applied if unspecified

DB Option Groups

- Oracle Application Express (APEX)
- SQL Server Transparent Data Encryption
- MySQL memcached support

RDS Configuration

RDS - AWS Console

Secure | https://us-east-2.console.aws.amazon.com/rds/home?region=us-east-2#parameter-groups:

Services Resource Groups

RDS Dashboard

Instances Clusters Reserved Instances Snapshots

Parameter Groups (selected)

Option Groups Subnet Groups Events Event Subscriptions

Notifications

Parameter Groups > mysql57parametergroup

Parameters Recent Events Tags

Filter: Search Parameters Cancel Editing Preview Changes Reset Parameters Save Changes

Name	Edit Values	Allowed Values
allow-suspicious-udfs	0, 1	
auto_generate_certs	0, 1	
auto_increment_increment	1-65535	
auto_increment_offset	1-65535	
autocommit	<engine-default>	
automatic_sp_privileges	<engine-default>	
avoid_temporal_upgrade	<engine-default>	
back_log	1-65535	
basedir	/rdsdbbin/mysql	
binlog_cache_size	32768	4096-18446744073709547520
binlog_checksum	<engine-default>	
binlog_error_action	<engine-default>	
binlog_format	MIXED	
binlog_group_commit_sync_delay	0-1000000	
binlog_group_commit_sync_no_delay_count	0-1000000	
binlog_gtid_simple_recovery	0, 1	
binlog_max_flush_queue_time	0-100000	
binlog_order_commits	<engine-default>	

Feedback English (US) © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Tel 52 95 20 79

RDS MySQL Parameter Group Options

RDS Monitoring

- Many metrics available for monitoring RDS Instances.

From AWS main console:

Endpoint: Not available yet ⓘ

The screenshot shows the AWS CloudWatch Metrics Metrics Details page for an RDS instance. On the left, there's a sidebar with icons for Alarms, Metrics, and Logs. The main area has two sections: "Alarms and Recent Events" and "Monitoring".

Alarms and Recent Events:

TIME (UTC-7)	EVENT
No Recent Events	

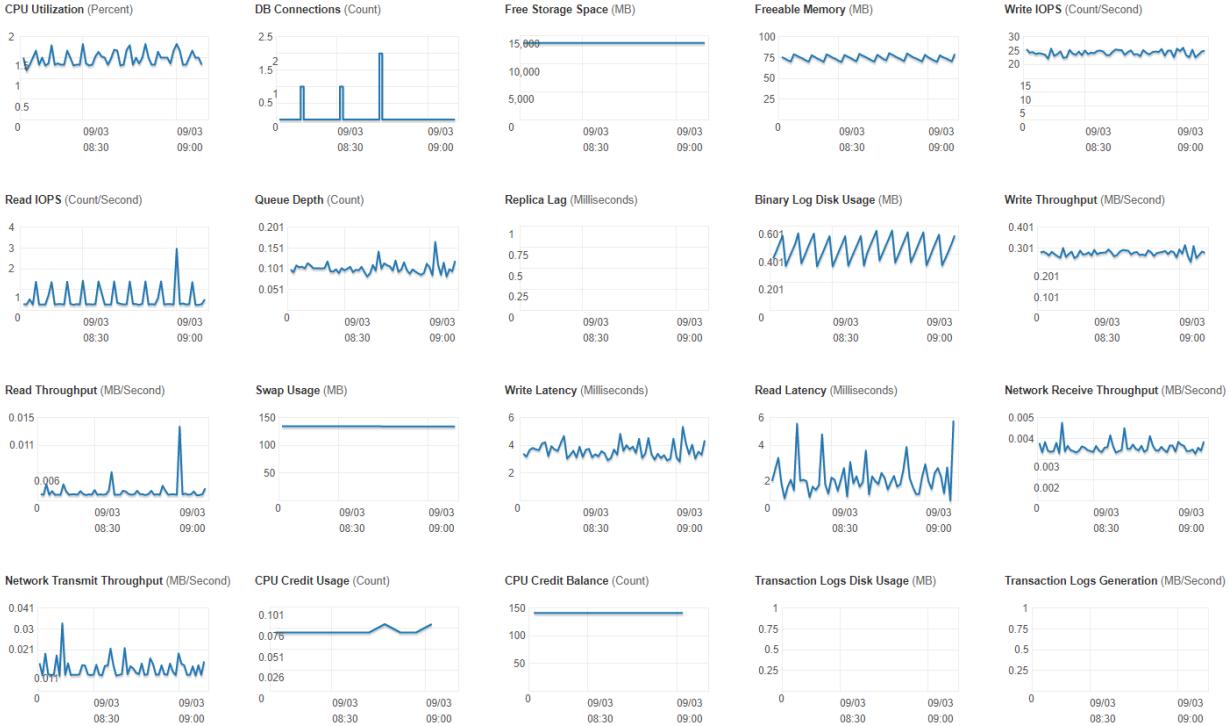
Monitoring:

	CURRENT VALUE	THRESHOLD	LAST HOUR		CURRENT VALUE	LAST HOUR
CPU	No Data	<div style="width: 100px; height: 10px; background-color: #ccc; border: 1px solid black;"></div>	<div style="width: 10px; height: 10px; background-color: red; border: 1px solid black;"></div>		Read IOPS	No Data
Memory	No Data	<div style="width: 100px; height: 10px; background-color: #ccc; border: 1px solid black;"></div>	<div style="width: 10px; height: 10px; background-color: red; border: 1px solid black;"></div>		Write IOPS	No Data
Storage	No Data	<div style="width: 100px; height: 10px; background-color: #ccc; border: 1px solid black;"></div>	<div style="width: 10px; height: 10px; background-color: red; border: 1px solid black;"></div>		Swap Usage	No Data

RDS Monitoring

Default Metrics

From a production database.



RDS Monitoring

Monitoring Charting Options

Show Multi-Graph View	Shows a summary of DB instance metrics available from Amazon CloudWatch. Each metric includes a graph showing the metric monitored over a specific time span.
Show Single Graph View	Shows a single metric at a time with more detail. Each metric includes a graph showing the metric monitored over a specific time span.
Show Latest Metrics View	Shows a summary of DB instance metrics without graphs. Full Monitoring View includes an option for full-screen viewing.
Enhanced Monitoring	Shows a summary of OS metrics available for a DB instance with Enhanced Monitoring enabled. Each metric includes a graph showing the metric monitored over a specific time span.

RDS Monitoring

Log Monitoring

- Log files are available from:
 - RDS Console
 - CLI
 - RDS API

The screenshot shows the AWS RDS console interface. On the left, a sidebar menu lists various RDS resources: Instances, Clusters, Reserved Instances, Snapshots, Parameter Groups, Option Groups, Subnet Groups, Events, Event Subscriptions, and Notifications. A large blue arrow points from the 'Logs' section of the main content area down towards the log table. The main content area displays the 'DB Instances > testdb' page. It has tabs for 'Details' and 'Recent Events & Logs'. The 'Recent Events & Logs' tab is selected, showing a table titled 'Most Recent Events' with three entries:

Time	Source	System Notes
September 3, 2017 at 9:11:35 AM UTC-7	testdb	Finished DB Instance backup
September 3, 2017 at 9:09:50 AM UTC-7	testdb	Backing up DB instance
September 3, 2017 at 9:09:31 AM UTC-7	testdb	DB instance created

Below this, a link 'see more events' is visible. The 'Logs' section contains a table with three log entries:

Name	Last Written	Size	view	watch	download
error/mysql-error-running.log	September 3, 2017 at 9:09:59 AM UTC-7	1.8 kB	view	watch	download
error/mysql-error.log	September 3, 2017 at 9:15:00 AM UTC-7	0 B	view	watch	download
mysqlUpgrade	September 3, 2017 at 9:09:00 AM UTC-7	3.4 kB	view	watch	download

At the bottom of the logs table, it says 'Viewing 3 of 3 Logs'. The footer of the browser window shows the URL 'https://us-east-2.console.aws.amazon.com/rds/home?region=us-east-2#dbinstance:id=testdb', the IP address '52.95.16.89', the language 'English (US)', and the copyright notice '© 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved.' along with links for 'Privacy Policy' and 'Terms of Use'.

RDS Monitoring

The screenshot shows the AWS RDS console interface. On the left, there's a sidebar with navigation links: Services, Resource Groups, Instances, Clusters, Reserved Instances, Snapshots, Parameter Groups, Option Groups, Subnet Groups, Events, Event Subscriptions (which is currently selected), and Notifications. The main content area is titled "Watching Log: error/mysql-error-running.log (1.8 kB)". It displays a log file with the following content:

```
text. background: black
2017-09-03 16:08:58 3342 [Note] Plugin 'FEDERATED' is disabled.
2017-09-03 16:08:58 3342 [Note] InnoDB: Using atomicity to ref count buffer pool pages
2017-09-03 16:08:58 3342 [Note] InnoDB: The InnoDB memory heap is disabled
2017-09-03 16:08:58 3342 [Note] InnoDB: Mutexes and rw_locks use GCC atomic builtins
2017-09-03 16:08:58 3342 [Note] InnoDB: Memory barrier is not used
2017-09-03 16:08:58 3342 [Note] InnoDB: Compressed tables use zlib 1.2.3
2017-09-03 16:08:58 3342 [Note] InnoDB: Using Linux native AIO
2017-09-03 16:08:58 3342 [Note] InnoDB: Using CPU crc32 instructions
2017-09-03 16:08:58 3342 [Note] InnoDB: Initializing buffer pool, size = 597.0M
2017-09-03 16:08:58 3342 [Note] InnoDB: Completed initialization of buffer pool
2017-09-03 16:08:58 3342 [Note] InnoDB: Highest supported file format is Barracuda.
2017-09-03 16:08:58 3342 [Note] InnoDB: 128 rollback segment(s) are active.
2017-09-03 16:08:58 3342 [Note] InnoDB: Waiting for purge to start
2017-09-03 16:08:58 3342 [Note] InnoDB: 5.6.35 started; log sequence number 1826510
2017-09-03 16:08:58 3342 [Warning] No existing UUID has been found, so we assume that this is the first time that this server has been started. Generating a new UUID: 316fde24-90c2-11e7-8223-0eeef4b40cb4.
2017-09-03 16:08:58 3342 [Note] Server hostname (bind-address): '*'; port: 3306
2017-09-03 16:08:58 3342 [Note] IPv6 is available.
2017-09-03 16:08:58 3342 [Note] - '::' resolves to '::'.
2017-09-03 16:08:58 3342 [Note] Server socket created on IP: '::'.
2017-09-03 16:08:58 3342 [Note] Event Scheduler: Loaded 0 events
2017-09-03 16:08:58 3342 [Note] /rdsdabbin/mysql/bin/mysqld: ready for connections.
Version: '5.6.35-log' socket: '/tmp/mysql.sock' port: 3306 MySQL Community Server (GPL)
-- END OF LOG --
```

Below the log content, a message says "Watching error/mysql-error-running.log, updates every 5 seconds." There is a "Close" button at the bottom right of the log viewer.

At the very bottom of the page, there are footer links: "52.95.16.89", "back", "English (US)", "© 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved.", "Privacy Policy", and "Terms of Use".

RDS Security

Security Groups

- Controls the access to a DB instance.
- Allows access from IP address ranges or Amazon EC2 instances.

Three Types

Type	Description
DB Security Group	Controls access to a DB instance that is not in a VPC. NOTE: This is legacy, only available for EC2 Classic accounts.
VPC Security Group	Controls access to a DB instance inside a VPC.
EC2 Security Group	Controls access to an EC2 instance and can be used with a DB instance.

RDS Security

IAM Permissions

Name	Description
AmazonRDSDirectoryServiceAccess	Allow RDS to access Directory Service Managed AD on behalf of the customer for domain-joined SQL Server DB instances.
AmazonRDSEnhancedMonitoringRole	Provides access to Cloudwatch for RDS Enhanced Monitoring.
AmazonRDSFullAccess	Provides full access to Amazon RDS via the AWS Management Console.
AmazonRDSReadOnlyAccess	Provides read only access to Amazon RDS via the AWS Management Console.
AWSQuickSightDescribeRDS	Allow QuickSight to describe the RDS resources.
RDSCloudHsmAuthorizationRole	Default policy for the Amazon RDS service role.

NOTE: These permissions are to access the service, not the data in the service.

RDS Best Practices

DB Instance Security Best Practices

- Use AWS Identity and Access Management (IAM) policies to assign permissions that determine who is allowed to manage RDS resources.
- Use security groups to control what IP addresses or EC2 instances can connect to your databases on a DB instance.

NOTE: you can use an SG as the source for an SG.

RDS Best Practices

DB Instance Security Best Practices

- Use Secure Socket Layer (SSL) connections with DB instances.
- Use RDS encryption to secure your RDS instances and snapshots at rest. Use network encryption and transparent data encryption with Oracle DB and MySQL instances.
- Use security features of your DB engine to control who can log in to the databases on a DB instance.

RDS Best Practices

- Monitor your memory, CPU, and storage usage in CloudWatch.
 - Set up notifications when usage patterns change or approach capacity of your deployment.
 - Can automate this process
 - Check storage utilization, scale
 - Handled by Aurora automatically.
- Scale up your DB instance when you are approaching storage capacity limits. Enable automatic backups and set the backup window to occur during the daily low in write IOPS.

RDS Best Practices

- Set a time-to-live (TTL) value of less than 30 seconds if connecting from app with DNS names.
- Test failover for your DB instance.
 - How long does the process take?
 - Ensure that the application can automatically connect to the new DB instance.

RDS Overview

RDS Demo

A large, light-gray circular icon containing a white right-pointing triangle, resembling a play button or a video thumbnail.

AWS Certified Developer (Associate) Crash Course

DynamoDB

DynamoDB Overview

- Managed NoSQL Database
- Fault tolerant design distributed across AZs,
 - Hands free
- Scale up/down without downtime
 - Not quick.
- Can be connected to a VPC through a VPC endpoint.
- Accessible through SSL/TLS API endpoints
- Time-to-live Data Expiration

DynamoDB Overview

Core Components

Primary Key Uniquely identifies the item

Attributes Key/Value pairs describing the **item**
 Similar to Columns

Item Contains Multiple **Attributes**
 No limit on the number of **items** in a database
 Similar to Rows

Table Collection of Data
 Contains Multiple **Items**

DynamoDB Overview

Naming Rules

Table, Item and Attribute names must conform to Naming Rules

- All names must be encoded using UTF-8
- Case-sensitive
- Table names and index names must be between 3 and 255 characters long, valid characters:
 - a-z
 - A-Z
 - 0-9
 - _ (underscore)
 - - (dash)
 - . (dot)
- Attribute names must be between 1 and 255 characters long.

DynamoDB Overview

Reserved Words

There are currently 573 reserved words in DynamoDB that cannot be used as attribute names.

Examples:

ABORT	ABSOLUTE	ACTION	ADD	AFTER
AGENT	ALL	ALLOCATE	ALTER	ANALYZE
AND	ANY	ARCHIVE	ARE	ARRAY
AS	ASC	ASCII	AT	ATOMIC

ON AND ON AND ON...

<http://docs.aws.amazon.com/amazondynamodb/latest/developerguide/ReservedWords.html>

DynamoDB Overview

DynamoDB Access

Access to DynamoDB is available through:

- DynamoDB Console
- AWS CLI
- AWS API

The screenshot shows the AWS DynamoDB console interface. At the top, there's a navigation bar with 'Services', 'Resource Groups', and a search bar. Below it, a table lists existing tables, with 'TestTable' being the only one shown. The table has columns for Name, Status, Partition key, Sort key, Indexes, Total read capacity, Total write capacity, and Auto Scaling. The 'TestTable' row shows 'Creating' as the status and 'TheKey (String)' as the partition key. Below the table, the 'Overview' tab of the 'TestTable' details page is selected. It displays Stream details (Stream enabled: No, View type: -, Latest stream ARN: -) and Table details (Table name: TestTable, Primary partition key: TheKey (String), Primary sort key: -, Time to live attribute: -, Table status: Creating, Creation date: September 3, 2017 at 9:45:46 AM UTC-7, Provisioned read capacity units: 5 (Auto Scaling Disabled), Provisioned write capacity units: 5 (Auto Scaling Disabled), Last decrease time: -, Last increase time: -, Storage size (in bytes): 0 bytes, Item count: 0, Region: US East (Ohio), Amazon Resource Name (ARN): arn:aws:dynamodb:us-east-2:146868985163:table/TestTable). A note at the bottom states: 'Storage size and item count are not updated in real-time. They are updated periodically, roughly every six hours.' At the bottom of the page, there are links for Feedback, English (US), and a copyright notice.

NOTE: DynamoDB is accessed exclusively through API, no JDBC/ODBC.

DynamoDB Console Interaction

DynamoDB Tables can be managed and queried through the console.

TestTable [Close](#)

Overview Items Metrics Alarms Capacity Indexes Triggers Access control Tags

[Create item](#) [Actions ▾](#)

Scan: [Table] TestTable: TheKey ▾

Scan [Table] TestTable: TheKey ▾
+ Add filter
Start search

	TheKey	aField	someNumber1	foo
<input type="checkbox"/>	aKey2			bar
<input type="checkbox"/>	aKey1	aFieldValue	12	

DynamoDB – Console Interaction

Item Addition via Console

Create item

Tree ▾

Item {3}

- + someNumber1 Number : 12
- + aField String : aFieldValue
- + TheKey String : aKey1

Create item

Text ▾

1 1

2 "someNumber1": 12,
3 "aField": "aFieldValue",
4 "TheKey": "aKey1"
5 }

DynamoDB JSON

DynamoDB – Keys

- **Primary Key:**
 - Single Attribute
 - Partition Key, determines the storage partition
 - Must be unique across the table
- **Composite Primary Key**
 - Composed of two attributes
 - Partition Key + Sort Key
 - Partition keys can overlap, must have different sort values.
 - Same partition key value are stored together, in sorted order by sort key value.
- Partition key also known as its *hash attribute*.

DynamoDB – Scan vs Query

Query

- Searches only primary key attribute values
- Supports comparison operators on values to refine search.
- Fast

Scan

- Scans the entire table.
- Specify filters to refine the results returned to you, after the scan.
- Big table, not fast. Consider secondary index.

DynamoDB – Indexes

Secondary Indexes

- Data structure that contains a subset of attributes from a table.
- Can query secondary index.
- Tables can have multiple secondary indexes.
- Supports scanning
- Associated with only one table, base table.
- Alternate key created
- Include all or only projected attributes.
- Secondary index is auto-updated when base table is modified.

DynamoDB – Indexes

Global Secondary Index

- Index with partition key and a sort key different from base table.
- “Global” = queries can span all of data in the base table, across all partitions.

Local Secondary Index

- Index with the same partition key as the base table
- Different sort key.
- “local” = scoped to a base table partition that has the same partition key value.

DynamoDB – Read Consistency

DDB Tables span availability zones for redundancy, takes time to replicate writes.

Eventually Consistent Reads (Default)

- Read response might not reflect the results of a recently completed write operation.

Strongly Consistent Reads

- DynamoDB returns a response with the most up-to-date data, reflecting the updates from all prior write operations that were successful.

DynamoDB – Read Capacity Units

Read Capacity Units

“A **read capacity unit** represents **one strongly consistent read per second**, or **two eventually consistent reads** per second, for an item up to 4 KB in size.”

- DynamoDB rounds up
- Item larger than 4 KB? Divide by 4KB to determine RCU needed for one read.
 - Item 8KB == 2 strongly consistent reads, 1 eventually consistent

Math

- Strong Reads/Sec = RCU / (itemsize / 4KB)
- Eventual Reads/Sec = RCU / (itemsize / 8KB)

DynamoDB – Read Capacity Examples

Read Capacity Unit Examples:

- Table with 20 RCU and items < 4KB:
 - Perform 20 strongly consistent reads per second, or
 - 40 eventually consistent reads per second
 - Strong: $20_{\text{items/sec}} = 20_{\text{RCU}} / (4_{\text{KB}} / 4_{\text{KB}})$
 - Eventual: $40_{\text{items/sec}} = 20_{\text{RCU}} / (4_{\text{KB}} / 8_{\text{KB}})$
- Table with 24 RCU and 8KB > items < 12KB:
 - Perform 8 strongly consistent reads per second, or
 - 16 eventually consistent reads per second
 - Strong: $8_{\text{items/sec}} = 24_{\text{RCU}} / (12_{\text{KB}} / 4_{\text{KB}})$
 - Eventual: $16_{\text{items/sec}} = 24_{\text{RCU}} / (12_{\text{KB}} / 8_{\text{KB}})$

DynamoDB – Read Capacity Examples

How many RCU would you need on a table to read 10 items per second with eventual consistency, each item being 83KB in size?

DynamoDB – Read Capacity Examples

How many RCU would you need on a table to read 10 items per second with eventual consistency, each item being 83KB in size?

1. How many read capacity units are required for one strong read of one item?

DynamoDB – Read Capacity Examples

How many RCU would you need on a table to read 10 items per second with eventual consistency, each item being 83KB in size?

1. How many read capacity units are required for one strong read of one item?

Round 83KB up, $84 / 4 == 21$ read cap. units for strong.

DynamoDB – Read Capacity Examples

How many RCU would you need on a table to read 10 items per second with eventual consistency, each item being 83KB in size?

1. How many read capacity units are required for one strong read of one item?

Round 83KB up, $84 / 4 == 21$ read cap. units for strong.

2. Is it eventual consistency?

DynamoDB – Read Capacity Examples

How many RCU would you need on a table to read 10 items per second with eventual consistency, each item being 83KB in size?

1. How many read capacity units are required for one strong read of one item?

Round 83KB up, $84 / 4 == 21$ read cap. units for strong.

2. Is it eventual consistency?

Divide by 2, 10.5 read cap units for eventual read of one item.

DynamoDB – Read Capacity Examples

How many RCU would you need on a table to read 10 items per second with eventual consistency, each item being 83KB in size?

1. How many read capacity units are required for one strong read of one item?

Round 83KB up, $84 / 4 == 21$ read cap. units for strong.

2. Is it eventual consistency?

Divide by 2, 10.5 read cap units for eventual read of one item.

3. How many items?

DynamoDB – Read Capacity Examples

How many RCU would you need on a table to read 10 items per second with eventual consistency, each item being 83KB in size?

1. How many read capacity units are required for one strong read of one item?

Round 83KB up, $84 / 4 == 21$ read cap. units for strong.

2. Is it eventual consistency?

Divide by 2, 10.5 read cap units for eventual read of one item.

3. How many items?

$10_{\text{items}} * 10.5_{\text{rcuForOne}} == 105$ read capacity units needed.

DynamoDB – Read Capacity Examples

Another way to look at it:

IS = Item Size

OS = Operation per second (read or write)

Eventually consistent reads

$$\text{RCU} = \lceil \text{IS}/4 \rceil * \text{OS}/2$$

Previous Example:

$$\text{RCU} = \lceil \text{ceil}(83/4) * 10 \rceil / 2$$

$$\text{RCU} = (21 * 10) / 2$$

$$\text{RCU} = (210) / 2$$

$$\text{RCU} = 105$$

Strongly consistent reads

$$\text{RCU} = \lceil \text{IS}/4 \rceil * \text{OS}$$

Previous Example:

$$\text{RCU} = \lceil \text{ceil}(83/4) * 10 \rceil$$

$$\text{RCU} = 21 * 10$$

$$\text{RCU} = 210$$

DynamoDB – Write Capacity Units

Write Capacity Units

“A ***write capacity unit*** represents one write per second, for an item up to 1 KB in size.”

No Strong or Eventual Consistency to worry about.

$$\text{WCU} = \text{ceil}(\text{IS}/1) * \text{OS}$$

IS = Item Size

OS = Operation per second (read or write)

DynamoDB – Write Capacity Examples

What is WCU required to write 10 items of size 83KB per second?

$$\text{WCU} = \text{ceil}(IS/1) * OS$$

$$\text{WCU} = \text{ceil}(83/1) * 10$$

$$\text{WCU} = 83 * 10$$

$$\text{WCU} = 830$$

DynamoDB – Scaling

DynamoDB Auto Scaling (Enabled by default)

Automatic Throughput Capacity Management

- Uses the AWS Application Auto Scaling service
- Scales based on traffic patterns
- Works on both table and GSI
- Can scale down as well.
- **Scaling policy:**
 - Scale read and/or write capacity
 - Set min/max provisioned units
 - Target Utilization – percentage of consumed throughput at a point in time. Auto-scaling attempts to maintain target utilization.

DynamoDB – Cross Region Replication

DynamoDB Cross Region Replication, Global Tables

- Automatically replicates DynamoDB table data to another region.
- Supports multi-master writes
- Table must be empty to enable
 - Plan for it.
- Requires streams to be enabled.

DynamoDB – Streams

Streams

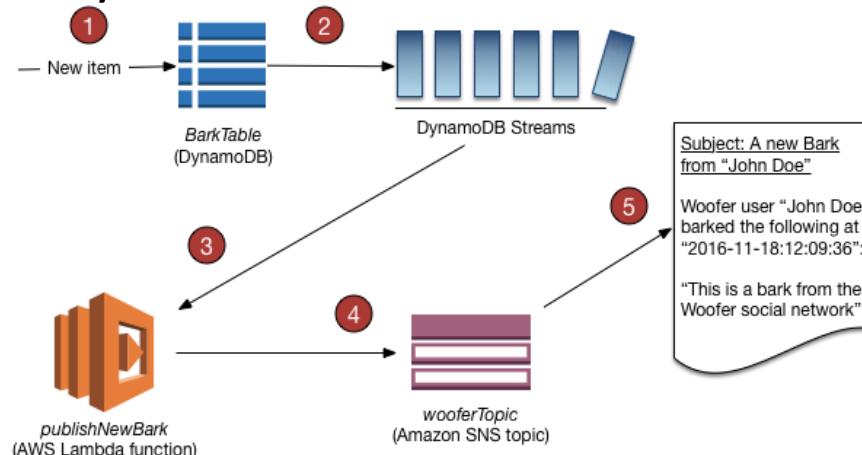
Near-real-time streaming of creates, updates, or deletes of data in a table.

- **Stream record:** information about a data modification to a single item in a DynamoDB table. Stream record can be:
 - Keys only
 - New image only
 - Old image only
 - Both new and old images

DynamoDB – Triggers

Triggers

- Execute Lambda function when data modified in table.
- Uses streams to send old/new image to Lambda function.
- AWS Lambda polls the stream and invokes your function synchronously when new stream records are encountered.



DynamoDB Security

There are no security groups with DynamoDB

Access is controlled through:

- Authentication
- Access Control

Authentication Via:

- AWS Account Root User
- IAM User with appropriate permissions
- IAM Role with appropriate permissions

DynamoDB Security

Access Control

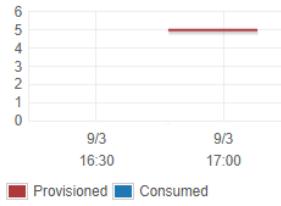
- Access enforced with permissions
- Identity-based Policies
 - Attach a permissions policy to a user or a group in your account
 - Attach a permissions policy to a role
 - Facebook, Google, Login with Amazon

DynamoDB Monitoring

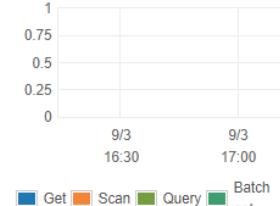
AWS Console monitoring similar to RDS Metrics

Capacity: table

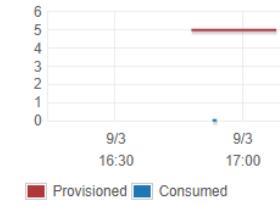
Read capacity (Units/Second - 1 min avg.)



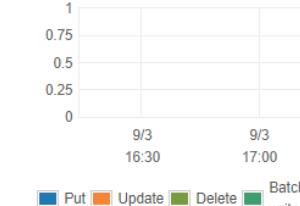
Throttled read requests (Count)



Write capacity (Units/Second - 1 min avg.)



Throttled write requests (Count)

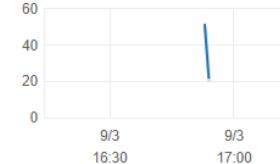


Latency

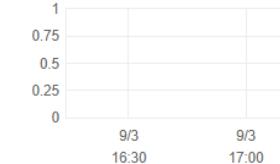
Get latency (Milliseconds)



Put latency (Milliseconds)



Query latency (Milliseconds)



Scan latency (Milliseconds)



DynamoDB Monitoring

DynamoDB Dashboard Shows:

- Recent Alerts
- Total Capacity
- Service Health

CloudWatch Shows:

- Current Alarms and Status
- Graphs of Alarms and Resources
- Service Health Status

DynamoDB Monitoring

Automated AWS Tools for DynamoDB Monitoring:

Amazon CloudWatch Alarms

- Metric over time monitoring
- Alerting with Amazon Simple Notification Service (Amazon SNS) topic

Amazon CloudWatch Logs

- Monitor, store, and access your log files

Amazon CloudWatch Events

- Match events and route them to one or more target functions or streams

AWS CloudTrail Log Monitoring

- Share log files between accounts
- Monitor CloudTrail log files in real time
- Write log processing applications in Java
- Validate that your log files have not changed after delivery

DynamoDB Best Practices

- Understand the differences between Relational Data Design and NoSQL
- Maintain as few tables as possible in a DynamoDB application *
- Keep related data together

* Source:

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/bp-general-nosql-design.html>

DynamoDB Best Practices

- Use sort order.
- Architect your data so the queries are distributed.
- Use global secondary indexes, if needed.
- Use DynamoDB Auto Scaling.

DynamoDB Best Practices

- Extensive list of Best Practices for DynamoDB from a developer perspective:

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/best-practices.html>

DynamoDB Overview

DDB Demo

A large, light-gray circular icon containing a white right-pointing triangle, resembling a play button or a start icon.

AWS Certified Developer (Associate) Crash Course

Messaging Services

Agenda

- Simple Queue Service
- Simple Notification Service



AWS Certified Developer (Associate) Crash Course

Simple Queue Service

SQS Overview

Simple

Pretty easy to use

Queue

Items Awaiting
Attention

Service

AWS Service

- Secure
- Durable
- Highly Available

- Scalable
- Reliable
- Standard & FIFO Queues

SQS Overview

Accessing SQS

- AWS Management Console
- AWS CLI
- AWS Tools for Windows PowerShell
- AWS SDKs
- Amazon SQS Query API

SQS Overview

- Managed distributed queue service
- Temporary repository for messages that are awaiting processing.
- Allows for decoupling distributed software systems and components.

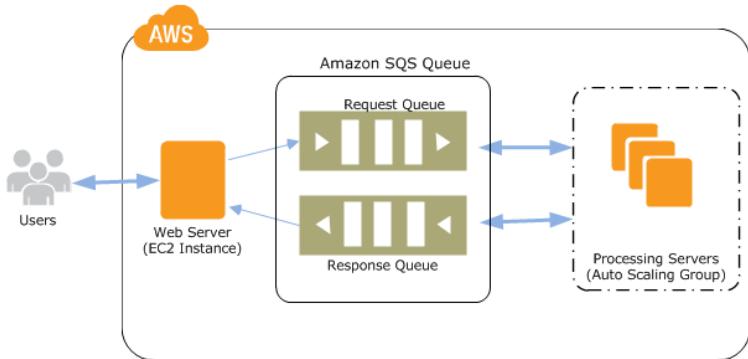


Image © Amazon

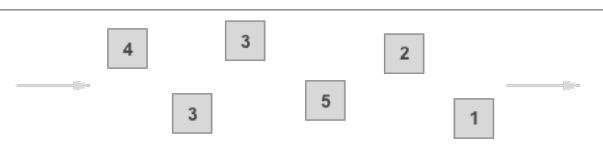
Offers:

- Dead Letter Queues
- Cost Allocation Tags

SQS Overview – Queues

Standard

- Available in all regions
- Unlimited Throughput
 - “Nearly unlimited” number of transactions per second (TPS) per action
e.g. ReceiveMessage, DeleteMessage
- At-Least-Once Delivery
 - A message is delivered at least once
 - More than one copy of a message is possible
- Best-Effort Ordering
 - Messages may be delivered out-of-order



FIFO

- N. VA, Ohio, Oregon, Ireland Regions
- High Throughput
 - 3,000 MPS – Batching
 - 300 MPS – No Batching
- Exactly-Once Processing
 - A message is delivered once and remains available until a consumer processes and deletes it. No duplicates
- First-In-First-Out Delivery
 - The order messages are sent and received is strictly preserved



SQS Overview

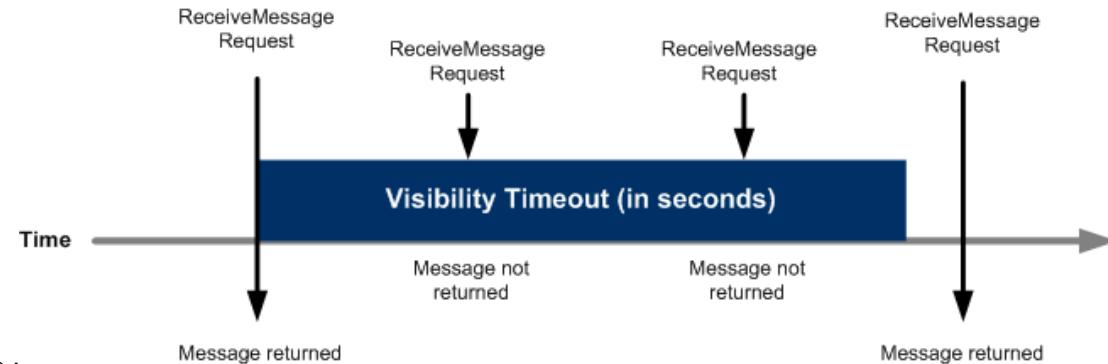
At-Least-Once Delivery

- Messages are copied to multiple servers when added to the queue.
- Server with a copy of the message may be unavailable when the message is received or deleted. “Rare”
- Message can be returned again.
- Applications should handle this message duplication gracefully.

SQS Overview

Visibility Timeout

- Started when a message is retrieved from the queue.
- If message not deleted before timer expiration, can be retrieved again.
- Consumer should process and delete messages from the queue before the timeout expires.
- Consumer can extend the timeout programmatically if still processing.



SQS Overview

Visibility Timeout

- Default: 30 Seconds
- Maximum: 12 Hours
- In-flight Message: a message that has been received but not deleted.
 - Max. In-flight:
 - Standard Queue: 120,000
 - FIFO Queue: 20,000

Message Retention Period

- The maximum amount of time a message will wait in the queue.
- Default: 4 Days
- Min/Max: 1 Minute / 14 Days

SQS Overview

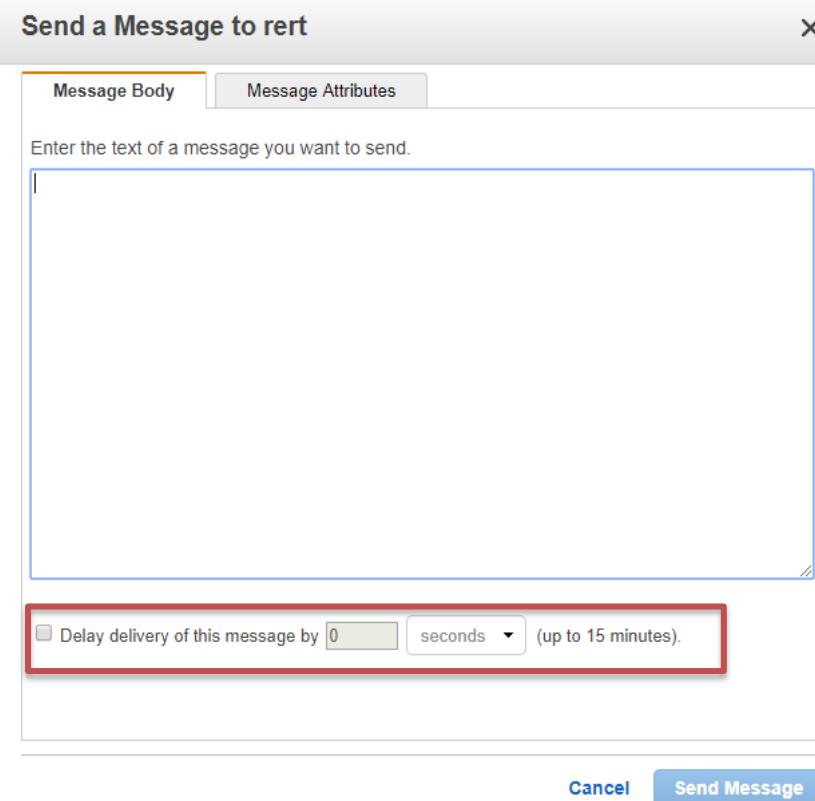
Message Size

- Default: 256 KB
- Min/Max: 1 / 256 KB

Delivery Delay (Queue Setting)

- Delay the first delivery of all messages added to this queue by X.
- Default: 0 Seconds (deliver immediately)
- Min/Max: 0 Seconds / 15 Minutes

SQS Overview



Message Timers

- Specify an invisibility timer for a message added to a queue.
- Not supported on FIFO Queues.

SQS Overview

Dead Letter Queue

- Queue for messages that cannot be processed successfully.
- Helpful for debugging consumers.
- Configured with Redrive Policy
- If `Message:ReceiveCount > Queue:maxReceiveCount`
Then move the message to a dead-letter queue.
- Queue Maximum Receive Count: $1 \geq X \leq 1000$

SQS Overview

Cost Allocation Tags

- Key/Value Metadata on Queues
- Tied to AWS Billing
- Enable AWS account bill to include tag keys and values

Example:

Queue	Key	Value
QAQueue	QueueType	Testing
ProdQueue	QueueType	Production

SQS Overview

Server Side Encryption (SSE)

- SQS can encrypt messages upon queue reception.
- Permits transmission of sensitive data.
- Keys managed by Amazon Key Management Service (KMS)
- Encrypts the message body.
- Does NOT encrypt:
 - Queue metadata (queue name and attributes)
 - Message metadata (message ID, timestamp, and attributes)
 - Per-queue metrics
- Dead-letter queue movement does not change encryption state.

Note: SQS SSE is not available in China.

SQS Overview – Security

Authentication

- Permission to access and manage SQS is provided by IAM.

Access Control

- SQS has its own resource-based permissions system.
- Uses the same language as IAM.

SQS Overview

CloudWatch Monitoring

- SQS and CloudWatch are integrated.
- Collect, view, and analyze metrics for active SQS queues.
- Metrics configured are pushed to CloudWatch every 5 minutes.
- Supports both Standard and FIFO queues.

SQS Overview

CloudWatch SQS Metrics

- The following metrics are reported by CloudWatch.
- ApproximateAgeOfOldestMessage
- ApproximateNumberOfMessagesNotVisible
- NumberOfEmptyReceives
- NumberOfMessagesReceived
- SentMessageSize
- ApproximateNumberOfMessagesDelayed
- ApproximateNumberOfMessagesVisible
- NumberOfMessagesDeleted
- NumberOfMessagesSent

SQS Best Practices

- Tune visibility timeout based on processing time.
 - Process Time + Delete Time + Buffer
- Use retry and backoff logic in AWS SDKs to provide exponential backoff for request errors.
- Use Long Polling where possible to reduce SQS fees.

SQS Best Practices

- Use the redrive policy to capture messages that fail processing.
- Message enqueue time does not change when moved to the dead-letter queue. Ensure the dead-letter queue lifetime is sufficient.
- Avoid setting the number of maximum receives to 1 when you configure a dead-letter queue.

SQS Overview

SQS Demo

A large, light-gray circular icon containing a white play triangle, positioned on the left side of the slide.

AWS Certified Developer (Associate) Crash Course

Simple Notification Service

SNS Overview

Simple

Pretty easy to use.

Notification

Pub/Sub

Service

AWS Service

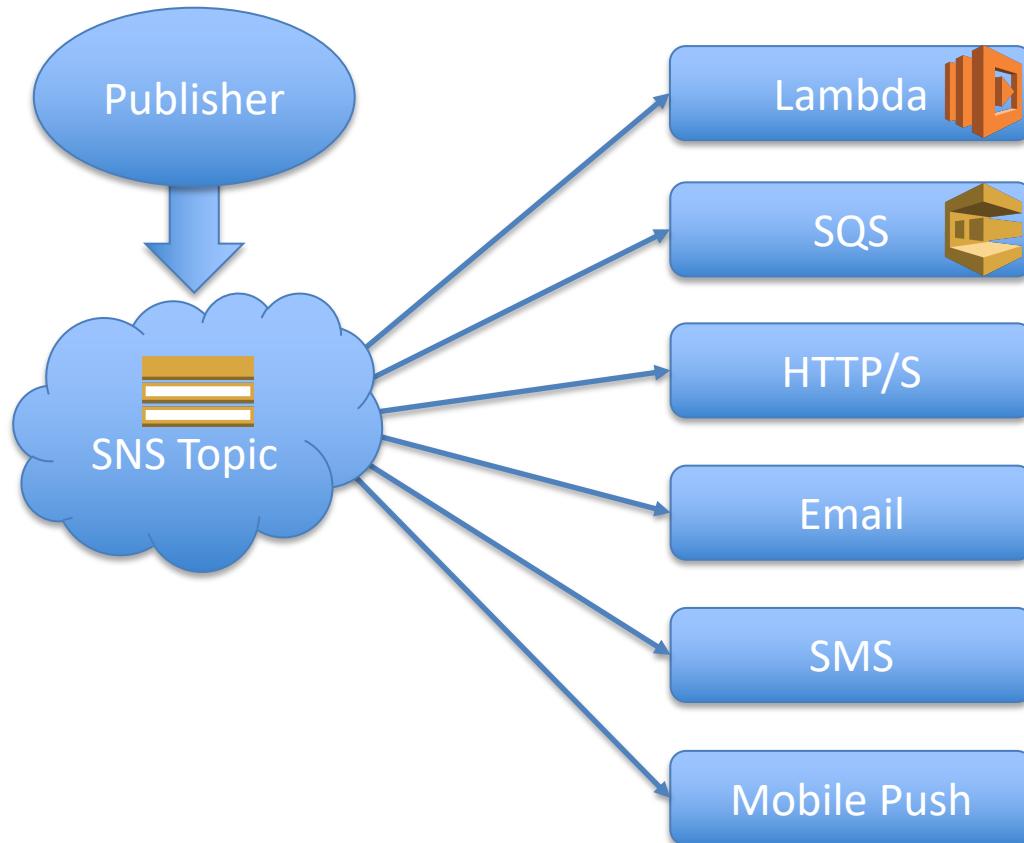
- Publishers / Providers
- Subscribers / Consumers

SNS Overview

Accessing SNS

- AWS Management Console
- AWS CLI
- AWS Tools for Windows PowerShell
- AWS SDKs
- Amazon SNS Query API

SNS Overview

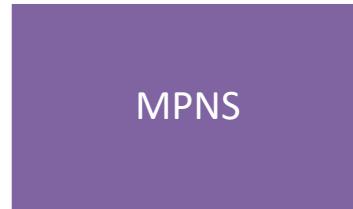
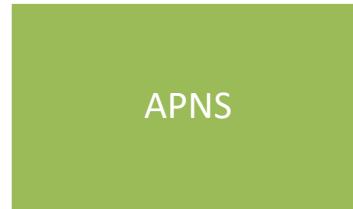
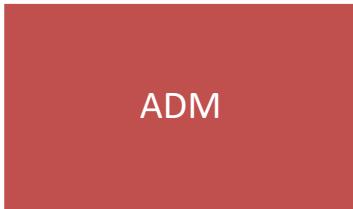


Note: Not all regions support SMS.

2020: Ireland, Frankfurt, N. VA, Oregon, Tokyo, Mumbai, Singapore, Sydney

SNS Overview

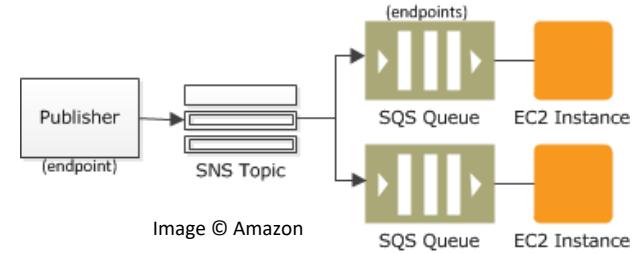
SNS Mobile Push Notification Services



SNS Overview – Common Use Cases

Fanout

- Message is sent to a topic
- Replicated and pushed to multiple:
 - Amazon SQS queues, HTTP endpoints, email addresses
- Parallel asynchronous processing



Example:

- Order placed -> publish to topic. Two SQS queues subscribed.
- EC2 instance handle the processing or fulfillment of the order
- EC2 instance attached to a data warehouse for analysis of all orders received.

SNS Overview – Common Use Cases

Application and System Alerts

- Many AWS Services use SNS for monitoring events
- Commonly used to send SMS to administrators

Create Alarm X

You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you define.

To edit an alarm, first choose whom to notify and then define when the notification should be sent.

Send a notification to:

Take the action: Recover this instance (i)
 Stop this instance (i)
 Terminate this instance (i)
 Reboot this instance (i)

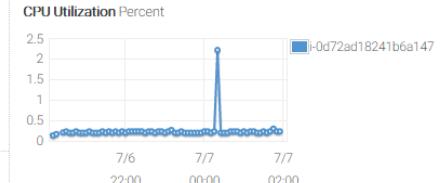
Whenever: of

Is: Percent

For at least: consecutive period(s) of

Name of alarm:

CPU Utilization Percent



Time	CPU Utilization Percent
22:00, 7/6	~0.1
00:00, 7/7	~2.2
02:00, 7/7	~0.1

SNS Overview – Common Use Cases

Push Email and Text Messaging

- Example: push targeted news headlines to subscribers by email or SMS.
- Upon receiving the email or SMS text, interested readers could then choose to learn more by visiting a website or launching an application.

SNS Overview – Common Use Cases

Mobile Push Notifications

- Send messages directly to mobile apps.

Example:

- Use Amazon SNS for sending notifications to an app, indicating that an update is available.
- The notification message can include a link to download and install the update.

SNS Overview

Message Durability

- SNS provides durable storage of all messages that it receives.
NOTE: Once the message is sent, it's removed.
- Publish requests are stored as multiple copies to disk.
- Stored to multiple, in-region, AZs prior to API acknowledgement receipt of request.

SNS Overview

Delivery Status

- Log delivery status of the following endpoints:
 - Application
 - HTTP
 - Lambda
 - SQS
- Log entries will be sent to CloudWatch Logs

SNS Overview

Filter Policies

- **Default:** subscriber receives all messages to topic.
- **Filter:** subscriber receives only messages that match filter.
 - Filter is a JSON object applied to a subscriber of a topic.
- Matches on **Message Attributes** in a message.

SNS Overview – Filter Policy Example

Message

```
{  
    "Type" : "Notification",  
    "MessageId" : "e3c4e17a-819b-5d95-a0e8-b306c25afda0",  
    "TopicArn" : "arn:aws:sns:us-east-1:111122223333:MySnsTopic",  
    "Message" : message body with transaction details . . .  
    "Timestamp" : "2017-11-07T23:28:01.631Z",  
    "SignatureVersion" : "1",  
    "Signature" : signature . . .  
    "UnsubscribeURL" : unsubscribe URL . . .  
    "MessageAttributes" : {  
        "customer_interests" : {"Type":"String.Array","Value":"[\"soccer\", \"rugby\"]"},  
        "store" : {"Type":"String","Value":"example_corp"},  
        "event" : {"Type":"String","Value":"order_placed"},  
        "price_usd" : {"Type": "Number", "Value": 210.75}  
    }  
}
```

Filter Policy

```
{  
    "store": ["example_corp"],  
    "event": [{"anything-but": "order_cancelled"}],  
    "customer_interests": ["rugby", "football", "baseball"],  
    "price_usd": [{"numeric": [">=", 100]}]  
}
```

SNS Overview

Lambda Subscription

- Lambda function can subscribe to a topic
- Function invoked with the payload of the published message
- Message payload is an input parameter.

Function can:

- Manipulate the information in the message
- Publish the message to other SNS topics
- Send the message to other AWS services

SNS Overview

HTTP/S Subscription

- HTTP and/or HTTPS endpoints can subscribe to a topic
- Data is sent using POST verb
- With HTTPS:
 - Specify Server Name Indication (SNI)
 - Enable Basic and Digest Access Authentication
 - Added in URL, e.g. `https://user:password@domain.com`

SNS Overview

CloudWatch Monitoring

- SNS and CloudWatch are integrated.
- Collect, view, and analyze metrics for active SNS notifications.
- Metrics configured are pushed to CloudWatch every 5 minutes.
- Metrics are gathered on all topics considered **Active**.
- **Active:** Any topics that has activity within the last six hours.
- Free.

SNS Overview

CloudWatch SNS Metrics

- The following metrics are reported by CloudWatch.
- NumberOfMessagesPublished
- NumberOfNotificationsFailed
- NumberOfNotificationsFilteredOut-NoMessageAttributes
- PublishSize
- SMSSuccessRate
- NumberOfNotificationsDelivered
- NumberOfNotificationsFilteredOut
- NumberOfNotificationsFilteredOut-InvalidAttributes
- SMSMonthToDateSpentUSD

SNS Best Practices

- Do not use HTTP for Topic Subscriptions
- Do not allow HTTP via Policy for Topic Subscriptions.
- When allowing cross-account publish or subscription, limit it to authorized/known AWS accounts.

SNS Best Practices

- Do not allow “everyone” to publish or subscribe to a topic.

Edit topic policy

Basic view Advanced view

Allow these users to publish messages to this topic

Only me (topic owner)
 Everyone
 Only these AWS users

Comma-separated list of AWS account IDs.

Allow these users to subscribe to this topic

Only me (topic owner)
 Everyone
 Only these AWS users

Comma-separated list of AWS account IDs.

Only users with endpoints that match

examples: "*@example.com" or "http://example.com/*"

Using these delivery protocols

<input checked="" type="checkbox"/> HTTP	<input checked="" type="checkbox"/> HTTPS	<input checked="" type="checkbox"/> Email
<input checked="" type="checkbox"/> Email-JSON	<input checked="" type="checkbox"/> SMS	<input checked="" type="checkbox"/> Amazon SQS
<input checked="" type="checkbox"/> Application	<input checked="" type="checkbox"/> AWS Lambda	

Cancel Update policy

SNS Overview

SNS Demo



AWS Certified Developer
(Associate) Crash Course
Execution Services

Agenda

- API Gateway
- Lambda
- Elastic Beanstalk
- CloudFormation



AWS Certified Developer (Associate) Crash Course

Execution Services

API Gateway

API Gateway Overview



API Gateway

“Amazon API Gateway is an AWS service that enables developers to create, publish, maintain, monitor, and secure APIs at any scale.”

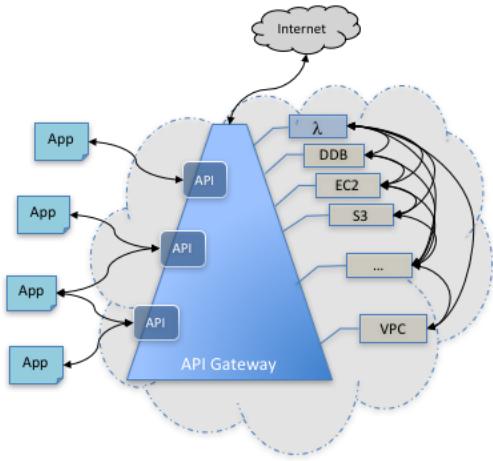
API Gateway Overview

App Developer vs. API Developer

- Amazon makes a distinction between the two
- **App Developer:**
 - Builds applications that call AWS services using the API Gateway
 - Does not need IAM credentials
 - **Unless API use requires permissions*
- **API Developer:**
 - Creates the API Gateway
 - Does need an IAM user account w/ appropriate permissions

API Gateway Overview

- API Gateway *can be* used as a **backplane** to connect AWS services
- Provides RESTful application programming interfaces (APIs)
- API Developer defines the API to enable access to AWS Cloud services



Example:

- Application call an API in API Gateway to upload user's photo
- Photo pushed to S3
- Lambda invoked to resize the image, saved in S3

API Gateway Overview

- API Gateway is part of AWS Serverless Infrastructure
- Tight integration with AWS Lambda
- API Gateway is the app-facing part of the AWS serverless infrastructure
- Lambda can be used to effect the intent of an app's API call

API Gateway Overview

API Gateway API:

- Integrates each API method with a backend endpoint
- Each backend endpoint is associated with an integration type

Integration Types

- Lambda
- HTTP
- Private
- Mock

API Gateway Overview

Lambda Integration

- **Proxy**
 - Preferred method, Simple:
Just point the API at a function, AWS configures the rest.
 - Input expressed as any combination of:
 - Request Headers
 - Path Variables
 - Query String Parameters
 - Body
- **Custom**
 - Legacy, greater developer involvement.
 - Can be used to pre-process request data.

API Gateway Overview

HTTP Integration

- **Proxy**
 - Client-submitted method request passed to backend.
 - Data Passed:
 - Request Headers
 - Path Variables
 - Query String Parameters
 - Body
 - Client and backend can interact directly after the API method is set up.
- **Custom**
 - More control of which data to pass between an API method and an API integration and how to pass the data.
 - Data mappings

API Gateway Overview

Private Integration

- Used to expose HTTP/HTTPS resources in an Amazon VPC for access by clients outside of the VPC.
- API with private integration can be used for open access or controlled access to private VPC resources.
- Controlled Access:
 - IAM permissions
 - Lambda authorizer
 - Amazon Cognito user pool

API Gateway Overview

Mock Integration

- Generate API responses from API Gateway directly
- Integration backend not needed.
- Unblock teams that need to work with an API before the project development is complete.
- API Landing Page
 - Example: Provide an overview of your API

API Gateway Overview

Controlling Access

- Amazon API Gateway Resource Policies
- IAM Permissions
- CORS
- Lambda Authorizers
- Amazon Cognito User Pools
- Client-Side SSL Certificates, Backend Authentication
- API Gateway Usage Plans

API Gateway Overview

Invoke

`https://{restapi_id}.execute-api.{region}.amazonaws.com/{stage_name} /`

- Obtain an API's Invoke URL in the API Gateway Console
- API Gateway Console -> Method Test
- Postman
- Generated SDKs
- AWS Amplify JavaScript Library
- Trace API Management and Invocation
 - CloudWatch

API Gateway Overview

Generated SDKs

API Gateway can generate SDKs for your API.

- Java SDK
- Android SDK
- JavaScript SDK
- Ruby SDK
- iOS SDK
 - Objective-C or Swift



AWS Certified Developer (Associate) Crash Course

Execution Services
Lambda

Lambda Overview



AWS Lambda

- Serverless Compute
- Analogue: anonymous lambda functions

Python:

```
def adder(x):  
    return lambda y: x + y  
adder = adder(5)  
adder(1)  
6
```

JavaScript:

```
var adder = function (x) {  
    return function (y) {  
        return x + y;  
    };  
};  
adder = adder(5);  
adder(1) == 6
```

Lambda Overview

- Run code without provisioning or managing servers.
- Code executed only when needed.
- Scales Automatically
- Pay for the compute time you consume
 - no charge when your code is not running.
- Code runs on highly-available compute infrastructure.
- Hands-off administration of the compute resources.

Lambda Overview

Automatic:

- Scaling
- Code monitoring
- Logging

Language Support:

Node.js

Python

Java

C#

Go

Lambda Functions can execute based on:

- Events
- Amazon API Gateway (HTTP call)
- API Calls
- Alexa

Lambda Overview

- Lambda Functions are launched into a container.
 - First execution latency
 - Container Freeze/Thaw on subsequent executions.
 - No reuse guarantee.
 - Containers run on Amazon Linux
- User specified max memory and execution time.
- Container provides 500MB of additional disk space
 - /tmp directory
 - Transient Cache
- Intra-region concurrent execution limit: 1000
 - Can request increase.

Lambda Event Sources

S3

DynamoDB

Kinesis
Streams

SNS

Simple
Email Svc.

Cognito

CloudFormation

CloudWatch
Logs

CloudWatch
Events

CodeCommit

AWS
Config

Alexa

Lex

API Gateway

Scheduled
Events

On-Demand

Lambda Event Sources

S3 Event Source

- Lambda functions can process S3 bucket events
 - object-created
Note: object-update is an object-created event.
 - object-deleted events

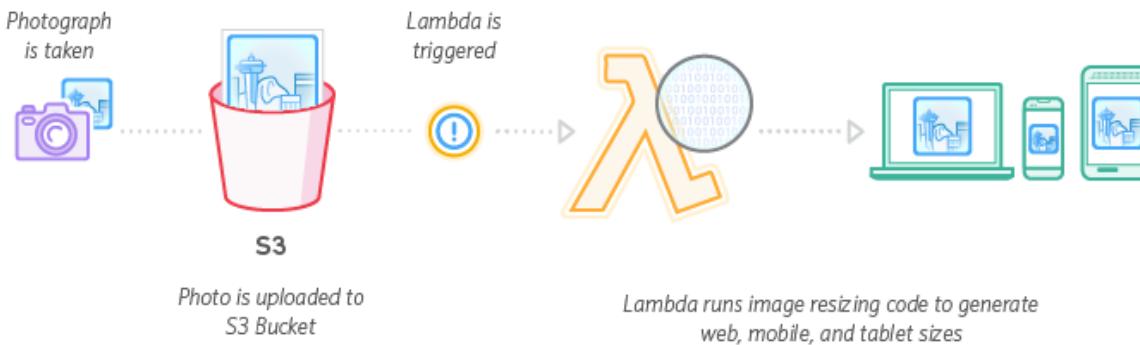


Image Source: Amazon

Lambda Event Sources

DynamoDB Event Source

- Lambda functions can execute based on updates to a DynamoDB table.
- Lambda polls the streams and function processes updates.



Image Source: Amazon

Lambda Event Sources

API Gateway Event Source

- Invoke a Lambda function over HTTPS.
 - Define a custom REST API and endpoint.
 - Map individual API operations, such as GET and PUT, to specific Lambda functions.
 - API Gateway invokes Lambda function based on request.

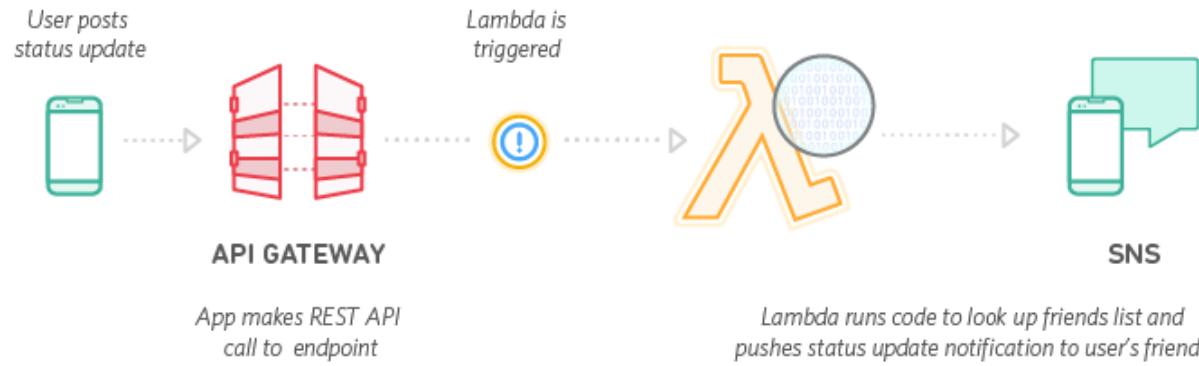


Image Source: Amazon

Lambda Scalability and Availability



AWS Lambda Limits

- AWS imposes limits on Lambda Functions
- Per invocation:

Resource	Limits
Memory allocation range	Minimum = 128 MB / Maximum = 3008 MB (with 64 MB increments) Memory Exceeded? Terminate
Ephemeral disk capacity ("tmp" space)	512 MB
Number of file descriptors	1,024
Number of processes and threads (combined total)	1,024
Maximum execution duration per request	300 seconds (5 minutes)
Invoke request body payload size (Req/Rep/synchronous) Response body also	6 MB
Invoke request body payload size (Event/asynchronous)	128 KB

Lambda Scalability and Availability



- 1,000 Concurrent Executions
 - Can be increased via support ticket
 - Concurrent execution limit control available at both account level and function level

Item	Default Limit
Lambda function deployment package size (compressed .zip/.jar file)	50 MB
Total size of all the deployment packages that can be uploaded per region	75 GB
Size of code/dependencies that you can zip into a deployment package (uncompressed .zip/.jar size). Note: 500MB available in /tmp	250 MB
Total size of environment variables set	4 KB

Lambda Scalability and Availability



Lambda Auto-scaling

- Dynamically scales capacity in response to increased traffic
- Subject to Account Level Concurrent Execution Limit
- Burst:
 - Lambda will immediately increase your concurrently executing functions by a predetermined amount, region based
 - If not enough, increase by 500/minute until account safety limit is reached

Region	Immediate Concurrency Increase
Ireland, Oregon, N. Virginia	3000
Tokyo, Frankfurt	1000
All other regions	500

Lambda Scalability and Availability

AWS Lambda is available in all regions except Osaka.

NOTE: The AWS Osaka Region is a Local Region

Local Region:

- Isolated, fault-tolerant infrastructure design located within a single datacenter
- One Availability Zone
- Intended for use with the Tokyo Region
- Customers must request access through a sales representative

Lambda Security



Lambda Security

- Integrated with IAM
- Can communicate with VPC through endpoint
 - Security Groups
 - Network ACLs
 - NOTE: Lambda functions provide access only to a single VPC
- SOC, HIPAA, PCI, ISO compliant
<https://aws.amazon.com/compliance/programs/>

Lambda Security



Lambda IAM Policies & Roles

Name	Description
AWSLambdaExecute	Provides Put, Get access to S3 and full access to CloudWatch Logs.
AWSLambdaFullAccess	Provides full access to Lambda.
AWSLambdaInvocation-DynamoDB	Provides read access to DynamoDB Streams.
AWSLambdaKinesisExecutionRole	Provides list and read access to Kinesis streams and write permissions to CloudWatch Logs.
AWSLambdaReadOnlyAccess	Provides read only access to Lambda, S3, DynamoDB, CloudWatch Metrics and Logs.
AWSLambdaReplicator	Grants Lambda Replicator necessary permissions to replicate functions across regions.

Lambda Security

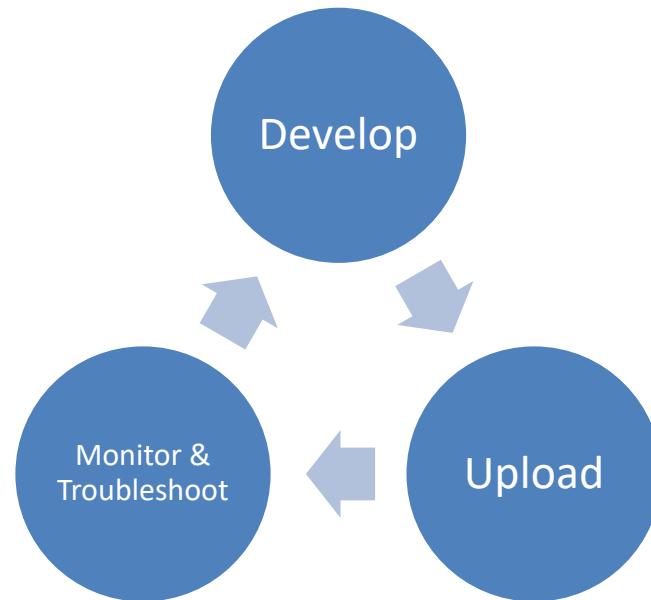


Lambda IAM Policies & Roles

Name	Description
AWSLambdaRole	Default policy for AWS Lambda service role.
AWSLambdaSQSQueueExecutionRole	Provides receive message, delete message, and read attribute access to SQS queues, and write permissions to CloudWatch Logs.
AWSLambdaVPCAccessExecutionRole	Provides minimum permissions for a Lambda function to execute while accessing a resource within a VPC - create, describe, delete network interfaces and write permissions to CloudWatch Logs.

Lambda Functions

Lambda Function Lifecycle



Lambda Functions

Developing Your Function

Which language?

- Node.js
- Python
- C#
- Java
- Go

External dependencies?

- AWS Lambda provides some libraries.
- You can upload others, including binaries.

Lambda Functions

Development Tools

Node.js

- Lambda Console
- Visual Studio (Plug-in)
- Your own environment.

Python / Go

- Lambda console
- Your own environment.

C#

- Visual Studio (Plug-in)
- .NET Core
- Your own environment.

Java

- Eclipse with AWS Toolkit
- Your own environment.

Lambda Functions

Lambda Function Code Pattern

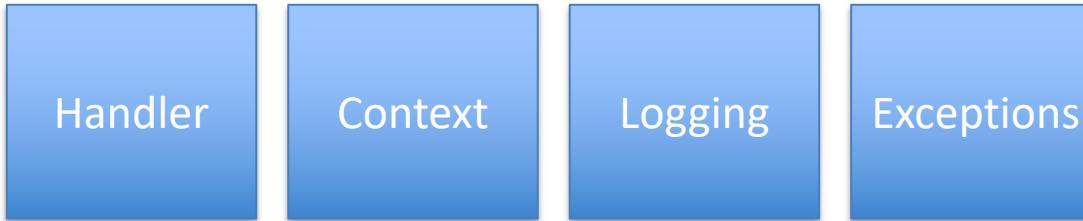
- Handler method that Lambda calls?
- Pass events to the handler?
- Log generation in CloudWatch Logs?
- Interaction with AWS Lambda runtime
 - Time remaining before timeout?
 - Exception Handling?

All of the answers are language specific.

Lambda Functions

Lambda Function Code Pattern

Referred to as Lambda Programming Model



Lambda Functions

Handler

- The function Lambda calls to start execution.
- Event data is passed as the first parameter.
- Handler can call other functions in your code.

Context Object

- Passed to the handler as the second parameter.
- Your code can interact with the Lambda runtime through the context object.
 - E.g. remaining execution time.

Lambda Functions

Logging

- Lambda function can contain logging statements.
- Logs are written to CloudWatch Logs.
- Log generation is language specific.
- **All stdout is logged.**

Exceptions

- Function must return result to Lambda.
- Success/Failure notification is language dependent.

Lambda Functions

We'll focus on the Node.js **programming model** here.

Handler

Context

Logging

Exceptions

All programming models are defined in the AWS Lambda documentation.

<http://docs.aws.amazon.com/lambda/latest/dg/programming-model-v2.html>

NOTE: The remainder of this lesson will review the programming model information with a Node.js Lambda function using AWS API Gateway as the trigger as it's the most common invocation.

Lambda Functions

Node.js Programming Model

Node.js runtimes:

- Node.js runtime v8.10
- Node.js runtime v6.10

Runtime specified at function creation.

Lambda Functions

Node.js Programming Model - Handler

```
exports.myHandler = function(event, context, callback) {  
    ...  
}
```

Handler

- Exported so Lambda can see it.
- Handler name specified when created.
- event – Event data from Lambda/Trigger
- context – Runtime information
- callback – Used to return data to the caller.

Lambda Functions

Node.js Programming Model – Event

Investigate Event Data Passed to Lambda Function

```
exports.myHandler = function(event, context, callback) {  
    var response = {  
        "isBase64Encoded": false,  
        "statusCode": 200,  
        "headers": {"Content-Type":"application/json"},  
        "body": JSON.stringify(event)  
    }  
    callback(null, response);  
}
```

Lambda Functions

```
{  
    "resource": "\/myFunc",  
    "path": "\/myFunc",  
    "httpMethod": "GET",  
    "headers": {  
        "Accept":  
            "text\/html,application\/xhtml+xml,application\/xml;q=0.9,image\/web  
p,image\/apng,*\/\*;q=0.8",  
        "Accept-Encoding": "gzip, deflate, br",  
        "Accept-Language": "en-US,en;q=0.8",  
        "CloudFront-Forwarded-Proto": "https",  
        "CloudFront-Is-Desktop-Viewer": "true",  
        "CloudFront-Is-Mobile-Viewer": "false",  
        "CloudFront-Is-SmartTV-Viewer": "false",  
        "CloudFront-Is-Tablet-Viewer": "false",  
        "CloudFront-Viewer-Country": "US",  
        "Host": "n8smv7ou0e.execute-api.us-east-2.amazonaws.com",  
        "upgrade-insecure-requests": "1",  
        "User-Agent": "Mozilla\/5.0 (Windows NT 10.0; Win64; x64)  
AppleWebKit\/537.36 (KHTML, like Gecko) Chrome\/60.0.3112.113  
Safari\/537.36",  
        "Via": "2.0 0302e8c462515ae281b72d9c815a5698.cloudfront.net  
(CloudFront)",  
        "X-Amz-Cf-Id": "3nNOdBj_zGJssmfh-  
LP8ESWUTaZDF51BJ3r6EyHuh8Hk8_lqsKiX_Q==",  
        "X-Amzn-Trace-Id": "Root=1-59adde97-708ed194361ffe976e20db2f",  
        "X-Forwarded-For": "x.x.x.x, a.a.a.a",  
        "X-Forwarded-Port": "443",  
        "X-Forwarded-Proto": "https"  
    },  
    "queryStringParameters": {  
        "value1": "value1",  
        "item1": "item1"  
    },  
    "pathParameters": null,  
    "stageVariables": null,  
    "requestContext": {  
        "path": "\/prod\/myFunc",  
        "accountId": "146868985163",  
        "resourceId": "0j3d0v",  
        "stage": "prod",  
        "requestId": "f519f306-91c6-11e7-ae39-f93cea0a06e4",  
        "identity": {  
            "cognitoIdentityPoolId": null,  
            "accountId": null,  
            "cognitoIdentityId": null,  
            "caller": null,  
            "apiKey": "",  
            "sourceIp": "x.x.x.x",  
            "accessKey": null,  
            "cognitoAuthenticationType": null,  
            "cognitoAuthenticationProvider": null,  
            "userArn": null,  
            "userAgent": "Mozilla\/5.0",  
            "user": null  
        },  
        "resourcePath": "\/myFunc",  
        "httpMethod": "GET",  
        "apiId": "n8smv7ou0e"  
    },  
    "body": null,  
    "isBase64Encoded": false  
}
```

} Data passed in.

} HTTP Verb Used

Lambda Functions

Node.js Programming Model – Context

Investigate Context Data

```
exports.myHandler = function(event, context, callback) {  
    var response = {  
        "isBase64Encoded": false,  
        "statusCode": 200,  
        "headers": {"Content-Type": "application/json"},  
        "body": JSON.stringify(context)  
    }  
    callback(null, response);  
}
```

Lambda Functions

Node.js Programming Model – Context

Context Data:

```
{  
  "callbackWaitsForEmptyEventLoop": true,  
  "logGroupName": "\/aws\/lambda\/myFunc",  
  "logStreamName": "2017\/09\/04\/[$LATEST]f7c71a4691774b0a9d7a235e75561b98",  
  "functionName": "myFunc",  
  "memoryLimitInMB": "128",  
  "functionVersion": "$LATEST",  
  "invokeid": "8c0c433a-91c8-11e7-ab6e-cb3c5423f2f1",  
  "awsRequestId": "8c0c433a-91c8-11e7-ab6e-cb3c5423f2f1",  
  "invokedFunctionArn": "arn:aws:lambda:us-east-2:146868985163:function:myFunc"  
}
```

Lambda Functions

Node.js Programming Model – Callback

Callback data when using AWS API Proxy Gateway must be in a particular format.

If not:

```
{ "message": "Internal server error" }
```

Valid Response:

```
{
  "isBase64Encoded": true|false,
  "statusCode": httpStatusCode,
  "headers": { "headerName": "HeaderValue", ... },
  "body": "..."
}
```

Lambda Functions

Node.js Programming Model – Callback

If data is not binary and no additional headers needed:

```
callback(null, {"statusCode": 200, "body": "results"});
```

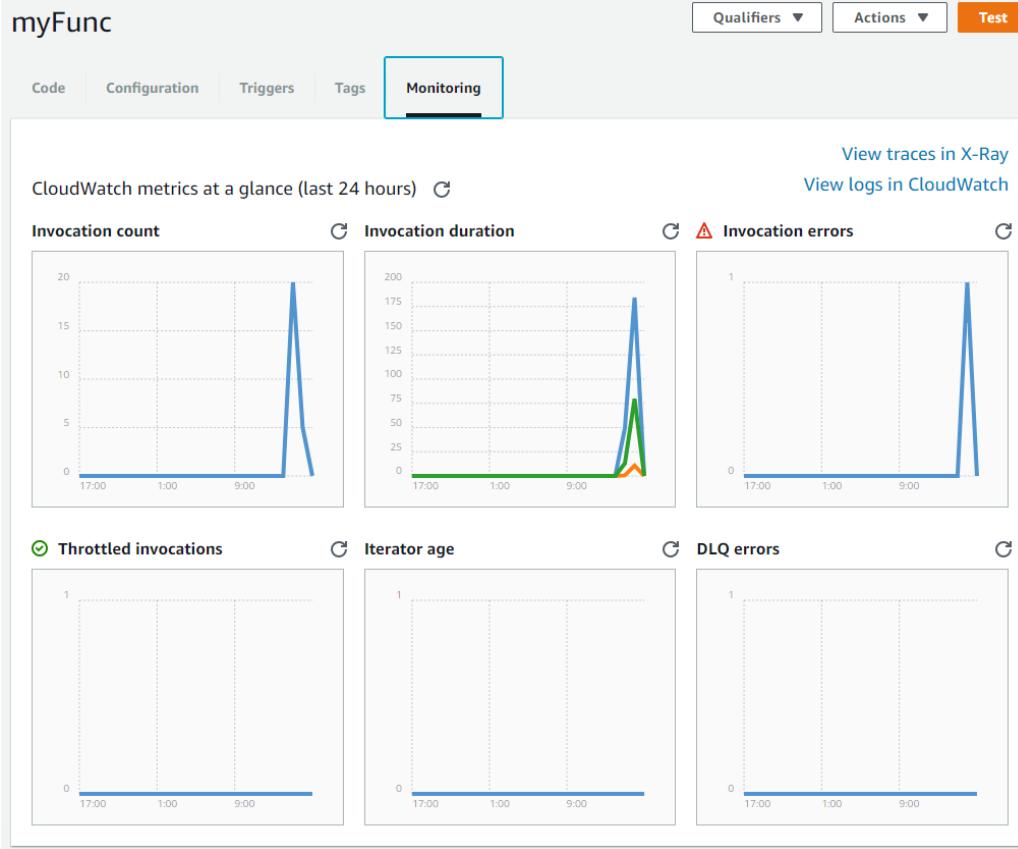
To throw an exception:

```
callback(new Error('internal server error'));
```

For client-side error:

```
callback(null, {"statusCode": 400, "body": "ErrorText"});
```

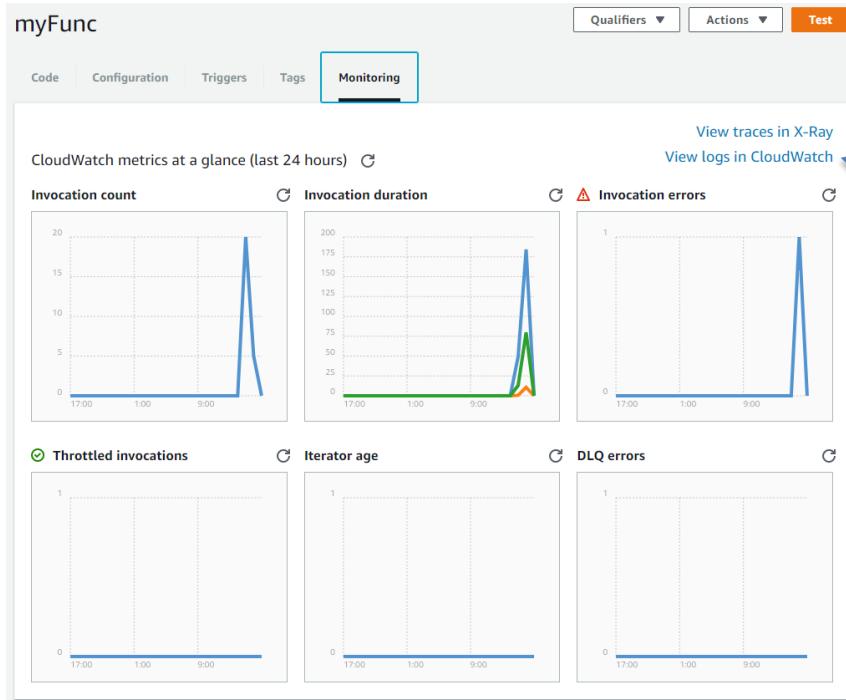
Lambda Functions



Monitoring

- AWS console provides monitoring metrics per lambda function.

Lambda Functions



Function Logs

- Logs can be accessed in CloudWatch.



Lambda Functions

The screenshot shows the AWS CloudWatch Management Console interface. The left sidebar navigation bar includes links for Services, Resource Groups, CloudWatch, Dashboards, Alarms, ALARM (highlighted in orange), INSUFFICIENT, OK, Billing, Events, Rules, Event Buses (NEW), Logs (highlighted in orange), and Metrics. The main content area displays log groups for the AWS Lambda service. A breadcrumb trail shows CloudWatch > Log Groups > /aws/lambda/myFunc > 2017/09/04[\$LATEST]f7c71a4691774b0a9d7a235e75561b98. A filter bar at the top allows filtering by Time (UTC +00:00) and Message, with a dropdown for time intervals (all, 30s, 5m, 1h, 6h, 1d, 1w, custom). Below the filter are buttons for Expand all, Row, Text, and a gear icon. The log entries are listed in a table with columns for Time (UTC +00:00) and Message. The first entry is a placeholder message: "No older events found at the moment. Retry.". Subsequent entries show log data for a Lambda function named "myFunc". The log entries include timestamp, log ID, and detailed JSON payload. The payload contains information such as resource name, path, HTTP method (GET), headers (including Accept, Accept-Encoding, Accept-Language, Cache-Control, CloudFront-Forwarded-Proto, CloudFront-Is-Desktop-Viewer, CloudFront-Is-Mobile-Viewer, CloudFront-Is-SmartTV-Viewer, CloudFront-Is-Tablet-Viewer, CloudFront-Viewer-Country, Host, User-Agent, Via, X-Amz-Cf-Id, X-Amzn-Trace-Id, X-Forwarded-For, X-Forwarded-Port, and X-Forwarded-Proto), query string parameters, path parameters, stage variables, and environment context.

CloudWatch Management

Services Resource Groups

CloudWatch Log Groups /aws/lambda/myFunc 2017/09/04[\$LATEST]f7c71a4691774b0a9d7a235e75561b98

Filter events

Time (UTC +00:00) Message

2017-09-04

No older events found at the moment. Retry.

23:26.59 2017-09-04T23:26:59.153Z 8c0c433a-91c8-11e7-ab6e-cb3c5423f2f1 Loading function

23:26.59 START RequestId: 8c0c433a-91c8-11e7-ab6e-cb3c5423f2f1 Version: \$LATEST

23:26.59 2017-09-04T23:26:59.157Z 8c0c433a-91c8-11e7-ab6e-cb3c5423f2f1 Received event: { "resource": "/myFunc", "path": "/myFunc", "httpMethod": "GET", "headers": { "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8", "Accept-Encoding": "gzip, deflate, br", "Accept-Language": "en-US,en;q=0.8", "Cache-Control": "max-age=0", "CloudFront-Forwarded-Proto": "https", "CloudFront-Is-Desktop-Viewer": "true", "CloudFront-Is-Mobile-Viewer": "false", "CloudFront-Is-SmartTV-Viewer": "false", "CloudFront-Is-Tablet-Viewer": "false", "CloudFront-Viewer-Country": "US", "Host": "n8sm7oud.execute-api.us-east-2.amazonaws.com", "Upgrade-Insecure-Requests": "1", "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36", "Via": "2.0 d8f42fc5558e4e9ebff8834baeb756.cloudfront.net (CloudFront)", "X-Amz-Cf-Id": "wI0jNpTgswCul0AK0v5lcl-qf4cB4spCmhItc9-FOWktwAu0C4oA=", "X-Amzn-Trace-Id": "Root=1-59ade142-1362e8105d0f5f7c7e798461", "X-Forwarded-For": "24.6.48.44, 54.239.134.12", "X-Forwarded-Port": "443", "X-Forwarded-Proto": "https" }, "queryStringParameters": { "value1": "value1", "item1": "item1" }, "pathParameters": null, "stageVariables": null, "requestContext": {} }

Feedback English (US)

© 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Tel 52.95.16.89

CloudWatch Lambda Function Logs

Lambda Functions

Lambda Layers

- Pull in additional code and content
- ZIP Archive
- Access the content of the layer during execution in the /opt directory
- Max of 5 layers
- Function + layers cannot exceed deployment package size of 250 MB.

Lambda Best Practices



General

- Test performance of your Lambda Function
 - Determine optimum memory size configuration
 - An increase in memory size triggers an equivalent increase in CPU available to your function
 - View memory usage in AWS CloudWatch Logs
- Load test your Lambda function, determine an optimum timeout value
 - Consider network calls to resources that may not handle Lambda's scaling

Lambda Best Practices



General

- Use most-restrictive permissions when setting IAM policies
- Familiarize yourself with AWS Lambda Limits
 - Consider: Payload size, file descriptors and /tmp space
- Delete Lambda functions that you are no longer using
 - Reduces Deployment Package Size
- Amazon Simple Queue Service event source: execution time should not exceed the Visibility Timeout value on the queue

Lambda Best Practices



Monitoring

- Use AWS Lambda Metrics and CloudWatch Alarms
 - Rather than creating or updating a metric from within your Lambda function code
- Leverage your logging library and AWS Lambda Metrics and Dimensions to catch app errors
 - e.g. ERR, ERROR, WARNING, etc.

Lambda Best Practices



Functions

- If you have to attach your Lambda function to a VPC:
 - Create dedicated Lambda subnets in your VPC
 - Can apply a custom route table for NAT Gateway traffic
 - Can dedicate address space to Lambda

Lambda Best Practices



Functions

- Separate the Lambda entry point from your core logic
 - Results in a more unit-testable function
- Use Execution Context Reuse to improve performance
 - Store external configurations or dependencies and reference them locally
- Limit the re-initialization of variables/objects on every invocation; instead use static initialization/constructor, global/static variables, and singletons

Lambda Best Practices



Functions

- Use Environment Variables to pass parameters
- Control dependencies in your function's deployment package
 - The AWS Lambda execution environment contains a number of libraries such the AWS SDK for the Node.js and Python runtimes
 - These libraries can be updated and might change your function's operation
- Minimize your deployment package to its runtime necessities
 - Reduces download and unpack times

Lambda Best Practices



Functions

- Reduce Lambda unpack time of deployment packages authored in Java by putting dependencies in a separate /lib directory
- Minimize the complexity of your dependencies
- Avoid using recursive code

Lambda Overview

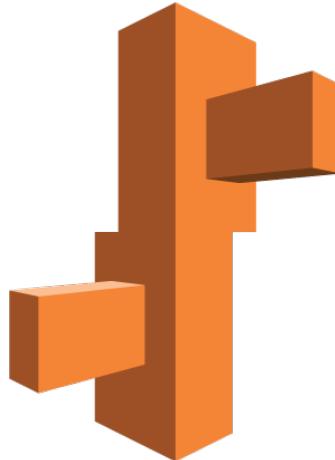
Lambda Demo



AWS Certified Developer (Associate) Crash Course

Execution Services
Elastic Beanstalk

Elastic Beanstalk Overview



Elastic Beanstalk
Rapid Application Development

User Manages:

- Code Development

AWS Manages:

- Capacity provisioning
- Load balancing
- Scaling
- Application health monitoring

Elastic Beanstalk Overview



Supports applications developed in/with:

- Go
- Java
- .NET
- Node.js
- Tomcat
- PHP
- Python
- Ruby
- Docker



Elastic Beanstalk Overview

Accessing Elastic Beanstalk

- AWS Management Console
- AWS CLI
- Elastic Beanstalk High-level CLI

Elastic Beanstalk Overview



Configure Test-env

Start from a preset that matches your use case or choose *Custom configuration* to unset recommended values and use the service's default values.

- Configuration presets Low cost (*Free Tier eligible*)
 High availability
 Custom configuration

Platform PHP 7.1 running on 64bit Amazon Linux/2.7.1 [Change platform configuration](#)

<p>Software</p> <p>Rotate logs: disabled (default) Log streaming: disabled (default) Environment properties: 0</p> <p>Modify</p>	<p>Instances</p> <p>EC2 instance type: t2.micro EC2 image ID: ami-b75d64d2 Root volume type: container default Root volume size (GB): container default Root volume IOPS: container default Security groups: none</p> <p>Modify</p>	<p>Capacity</p> <p>Environment type: single instance</p> <p>Modify</p>
<p>Load balancer</p> <p><i>This configuration does not contain a load balancer.</i></p>	<p>Rolling updates and deployments</p> <p>Deployment policy: All at once Rolling updates: disabled Health check: enabled</p> <p>Modify</p>	<p>Security</p> <p>Service role: autogenerated Virtual machine key pair: – Virtual machine instance profile: autogenerated</p> <p>Modify</p>
<p>Monitoring</p> <p>Health reporting system: Enhanced Health event log streaming: disabled</p> <p>Modify</p>	<p>Notifications</p> <p>Email address: –</p> <p>Modify</p>	<p>Network</p> <p><i>This environment is not part of a VPC.</i></p> <p>Modify</p>
<p>Database</p> <p>Engine: – Instance class: –</p>	<p>Tags</p> <p>Tags: none</p>	

Platform Configuration

Defines: infrastructure and software stack to be used for a given language environment.

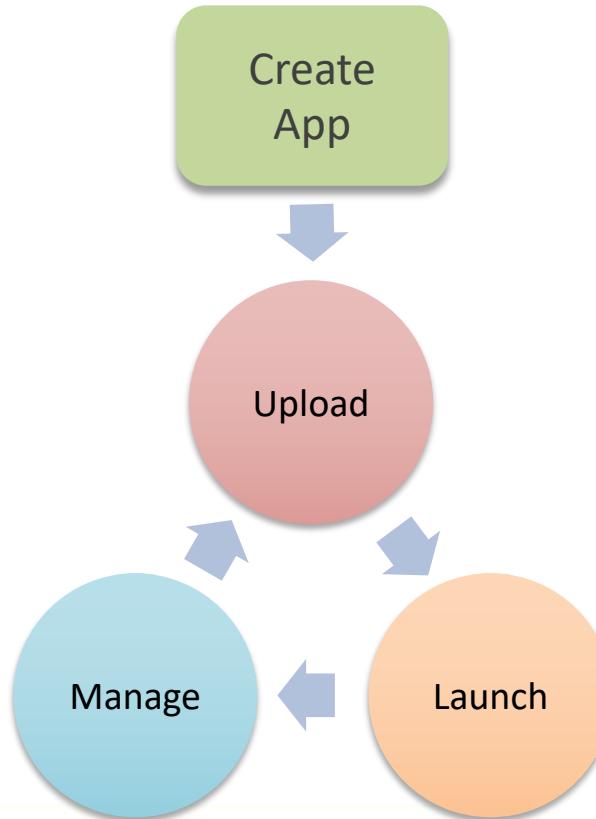
Elastic Beanstalk Overview

Application Lifecycle

- Create an application
 - Upload application source bundle (e.g. a Java .war)
 - Provide information about the application
- Elastic Beanstalk launches an environment:
 - Creates/configures the AWS resources needed to run your code
- After launch:
 - Manage your environment
 - Deploy new application versions

Elastic Beanstalk Overview

Application Lifecycle



Elastic Beanstalk Overview

Overview

Refresh



Health

Ok

Causes

Running Version

Sample Application

Upload and Deploy



Configuration

PHP 7.1 running on 64bit Amazon Linux/2.7.1

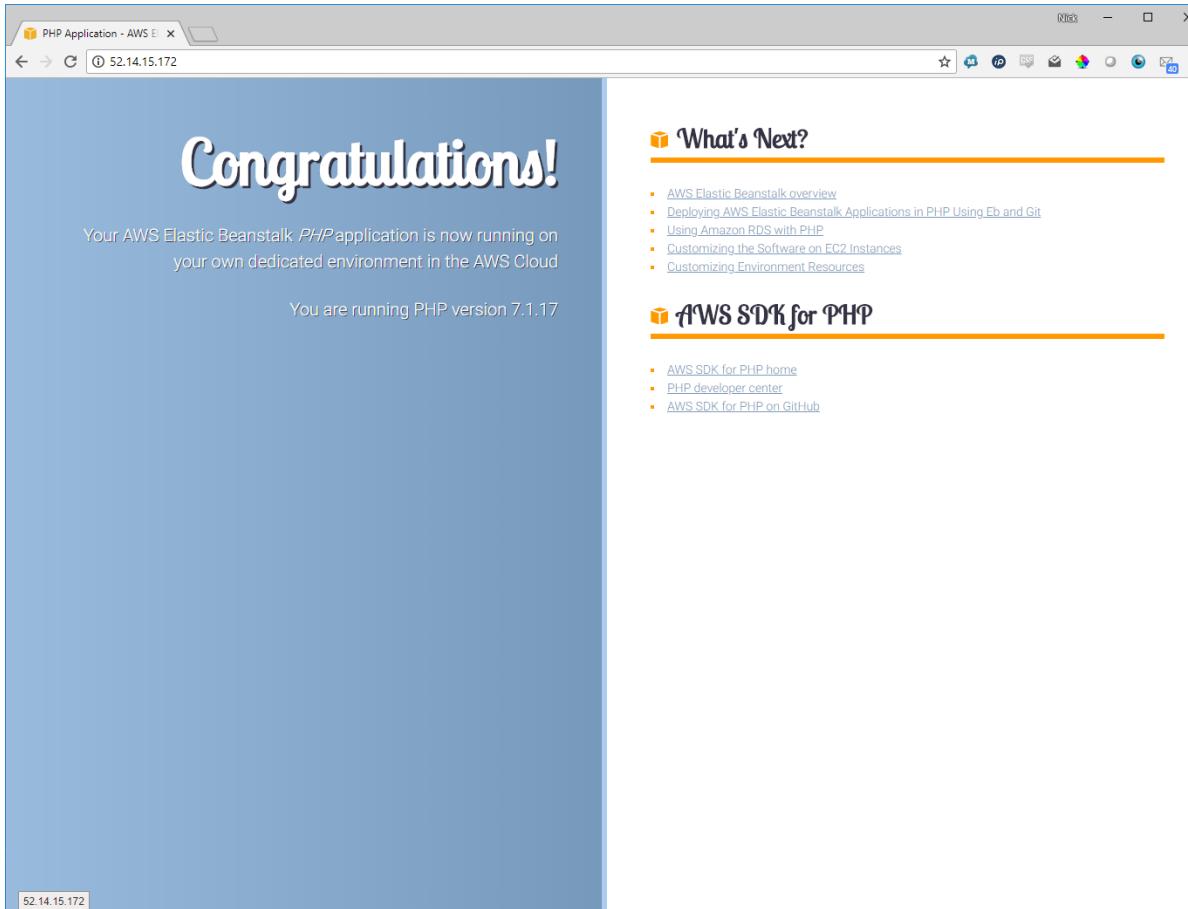
Change

Recent Events

Show All

Time	Type	Details
2018-07-08 08:33:44 UTC+0900	INFO	Successfully launched environment: Test-env
2018-07-08 08:33:12 UTC+0900	INFO	Environment health has transitioned from Pending to Ok. Initialization completed 5 seconds ago and took 2 minutes.
2018-07-08 08:32:32 UTC+0900	INFO	Waiting for EC2 instances to launch. This may take a few minutes.
2018-07-08 08:32:12 UTC+0900	INFO	Added instance [i-024ec4d13ae922b4b] to your environment.
2018-07-08 08:31:12 UTC+0900	INFO	Environment health has transitioned to Pending. Initialization in progress (running for 45 seconds). There are no instances.

Elastic Beanstalk Overview



Elastic Beanstalk Overview

All Applications > Test > Test-env (Environment ID: e-cx7mcbp9ht; URL: Test-env.ahc826fhe.us-east-2.elasticbeanstalk.com)

Actions ▾

Dashboard

Configuration

Logs

Health

Monitoring

Alarms

Managed Updates

Events

Tags

Enhanced Health Overview

Filter By ▾

Instance Actions ▾

Hide details

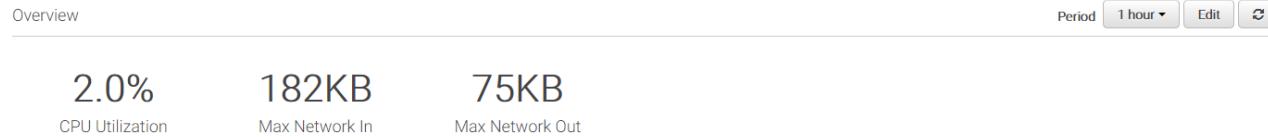


Auto refresh (0s)

	Server				Requests								Latency				Load Average		CPU Utilization				
	Instance ID	Status	Running ▾	Dep. ID	R/sec	2xx	3xx	4xx	5xx	P99	P90	P75	P50	P10	Load1	Load5	User%	Sys%	Idle%	I/O wait%			
Overall	Ok	N/A	N/A	—	—	—	—	—	—	—	—	—	—	—	N/A	N/A	N/A	N/A	N/A	N/A			
Total	1	Ok	1	Pending	0	Info	0	Unknown	0	No data	0	Warning	0	Degraded	0	Severe	0	0.00	0.01	0.3	0.1	99.6	0.0
i-024ec4d13ae922b4b	Ok	7 minutes	1	—	—	—	—	—	—	—	—	—	—	—	—	—	0.00	0.01	0.3	0.1	99.6	0.0	

NOTE: The instances instantiated are visible in the EC2 console.

Elastic Beanstalk Overview



Elastic Beanstalk Provided Monitoring

Elastic Beanstalk Best Practices

- Enable Enhanced Health Monitoring

Health reporting

Enhanced health reporting provides free real-time application and operating system monitoring of the instances and other resources in your environment. The [Environment health](#) custom metric is provided free with enhanced health reporting. Additional charges apply for each custom metric. For more information, see [Amazon CloudWatch Pricing](#).

System Enhanced Basic

CloudWatch Custom Metrics	Instance	Environment
ApplicationLatencyP10	ApplicationLatencyP10	ApplicationLatencyP10
ApplicationLatencyP50	ApplicationLatencyP50	ApplicationLatencyP50
ApplicationLatencyP75	ApplicationLatencyP75	ApplicationLatencyP75
ApplicationLatencyP85	ApplicationLatencyP85	ApplicationLatencyP85
ApplicationLatencyP90	ApplicationLatencyP90	ApplicationLatencyP90
ApplicationLatencyP95	ApplicationLatencyP95	ApplicationLatencyP95
ApplicationLatencyP99	ApplicationLatencyP99	ApplicationLatencyP99
ApplicationLatencyP99.9	ApplicationLatencyP99.9	ApplicationLatencyP99.9
ApplicationRequests2xx	ApplicationRequests2xx	ApplicationRequests2xx
ApplicationRequests3xx	ApplicationRequests3xx	ApplicationRequests3xx
ApplicationRequests4xx	ApplicationRequests4xx	ApplicationRequests4xx
ApplicationRequests5xx	ApplicationRequests5xx	ApplicationRequests5xx
ApplicationRequestsTotal	ApplicationRequestsTotal	ApplicationRequestsTotal
CPUIdle	CPUIdle	InstancesDegraded
CPUWait	CPUWait	InstancesInInfo
CPUIRQ	CPUIRQ	InstancesNoData
CPUNice	CPUNice	InstancesOK
CPUStoIRQ	CPUStoIRQ	InstancesPending
CPUSystem	CPUSystem	InstancesSevere
CPUUser	CPUUser	InstancesUnknown
InstanceHealth	InstanceHealth	InstancesWarning
LoadAverage1min		
LoadAverage5min		
RootFilesystemUtil		

Hold the Ctrl/Command key while clicking to select multiple metrics.

Elastic Beanstalk Best Practices

Enable CloudWatch Log Streaming

Instance log streaming to CloudWatch Logs

Configure the instances in your environment to stream logs to CloudWatch Logs. You can set the retention to up to ten years and configure Elastic Beanstalk to delete the logs when you terminate your environment.

Log streaming Enabled (Standard CloudWatch charges apply.)

Retention days

Lifecycle

Elastic Beanstalk Best Practices

Enable Managed Platform Updates

Modify managed updates

Managed platform updates

Enable managed platform updates to apply platform updates automatically during a weekly maintenance window that you choose. Your application stays available during the update process.

Managed updates Enabled

Weekly update window at : UTC

Update level

Instance replacement If enabled, an instance replacement will be scheduled if no other updates are available.

[Cancel](#) [Continue](#) [Apply](#)

EB Overview

EB Demo



AWS Certified Developer (Associate) Crash Course

Execution Services

CloudFormation

CloudFormation Overview



CloudFormation

“AWS CloudFormation enables you to create and provision AWS infrastructure deployments predictably and repeatedly.”

CloudFormation Overview

- Template-based creation and deletion of resources
- Collection of Resources: Stack
- Leverage:
 - Compute Services
 - Database Services
 - Storage Service
 - and many more...
- Underlying AWS infrastructure created for you
- IAM Access Controlled

CloudFormation Overview

Concepts

Templates

Stacks

Change Sets

StackSets

CloudFormation Overview

```
{  
    "AWSTemplateFormatVersion" : "2010-09-09",  
    "Description" : "A sample template",  
    "Resources" : {  
        "MyEC2Instance" : {  
            "Type" : "AWS::EC2::Instance",  
            "Properties" : {  
                "ImageId" : "ami-2f726546",  
                "InstanceType" : "t1.micro",  
                "KeyName" : "testkey",  
                "BlockDeviceMappings" : [  
                    {  
                        "DeviceName" : "/dev/sdm",  
                        "Ebs" : {  
                            "VolumeType" : "io1",  
                            "Iops" : "200",  
                            "DeleteOnTermination" : "false",  
                            "VolumeSize" : "20"  
                        }  
                    }  
                ]  
            }  
        }  
    }  
}
```

Concepts: Templates

- JSON or YAML text file
- Blueprint for the Stack
- Save locally or in S3

Note: local template uploads to S3 prior to execution.

CloudFormation Overview

Concepts: Stacks

- Collection of resources managed as a single unit.
- Create, update, and delete stacks.
- All resources defined by the stack's AWS CloudFormation template.
- Create Stack: Submit template -> AWS CloudFormation provisions all resources.
- Managed via:
 - Console
 - API
 - CLI

CloudFormation Overview

Concepts: Change Sets

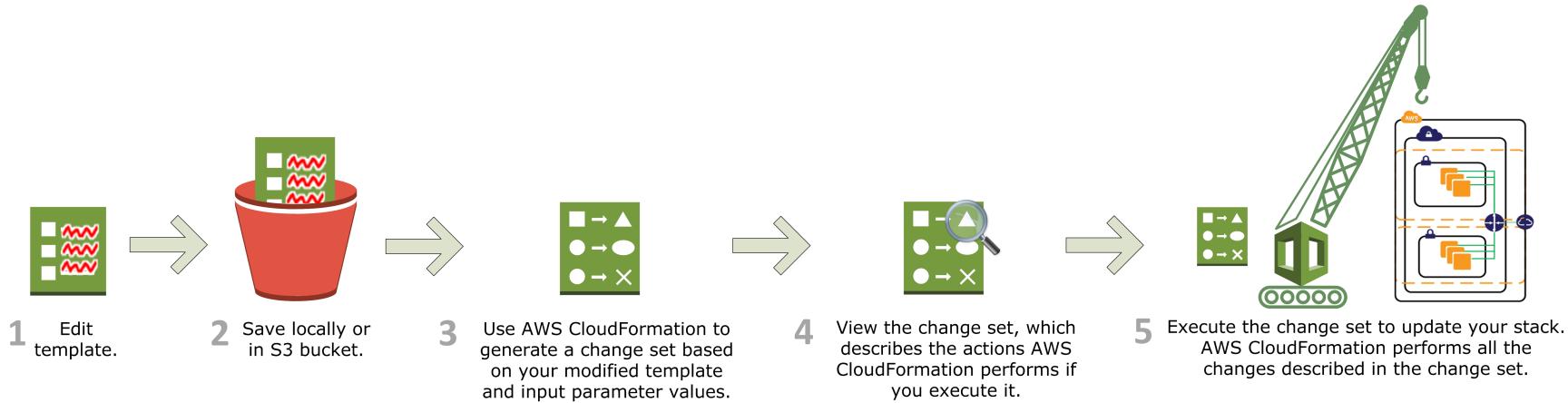
- Summary of your proposed changes to update a stack
- See how your changes might impact your running resources before implementation

Example:

- Change name of RDS instance
- CloudFormation will create a new database, delete the old
- Data in old database lost unless backed up
- Change set would tell you that database will be impacted

CloudFormation Overview

Concepts: Change Sets



CloudFormation Overview

Concepts: StackSets

- Enables creation of stacks in AWS accounts across regions using a single AWS CloudFormation template.
- Resources are defined by the stack set's AWS CloudFormation template.
- Create, update, or delete stacks in the target accounts and regions you specify
- Is a regional resource
 - Cannot see it or change it in other regions

CloudFormation Overview

Concepts: StackSets

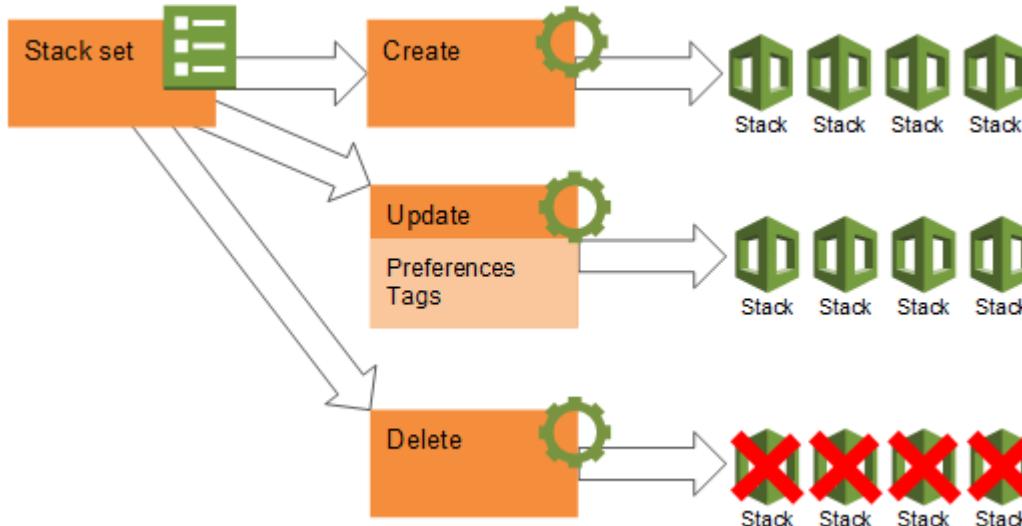


Image © Amazon

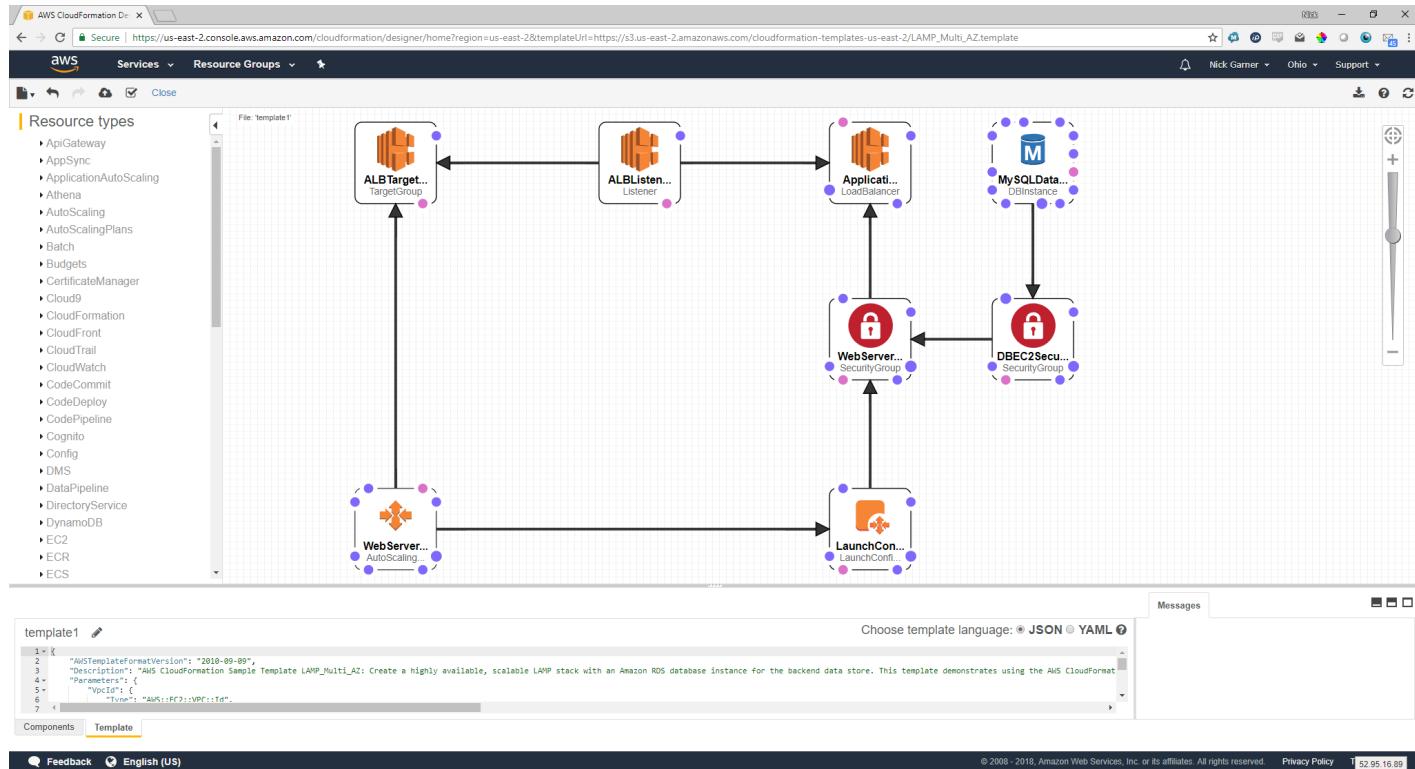
CloudFormation Overview

CloudFormation Designer

- Graphic representations of the resources in your template
- Simplifies template authoring and editing
- Enforces some basic relationships between resources
- GUI and Text at the same time

CloudFormation Overview

CloudFormation Designer



CloudFormation Best Practices

- Use IAM to control access
- Reuse templates to replicate Stacks in multiple environments
- Use Stack Sets for cross region
- Use nested stacks to reuse common template patterns
- Isolate templates to a single function.
- Use CF Templates with Service Catalog

CloudFormation Best Practices

- Do not embed credentials in your templates
- Use AWS-specific parameter types
- Use parameter constraints
- Use AWS::CloudFormation::Init to deploy software applications on Amazon EC2 instances
- Use the latest helper scripts
- Validate templates before using them

CloudFormation Best Practices

- Manage all stack resources through AWS CloudFormation
- Create change sets before updating your stacks
- Use stack policies
- Use AWS CloudTrail to log AWS CloudFormation calls
- Use revision controls to manage templates
- Update your Amazon EC2 Linux instances regularly



AWS Certified Developer (Associate) Crash Course

Developing in AWS

Agenda

- AWS SDKs
- CodeStar
- CodeCommit
- CodePipeline
- CodeBuild
- CodeDeploy



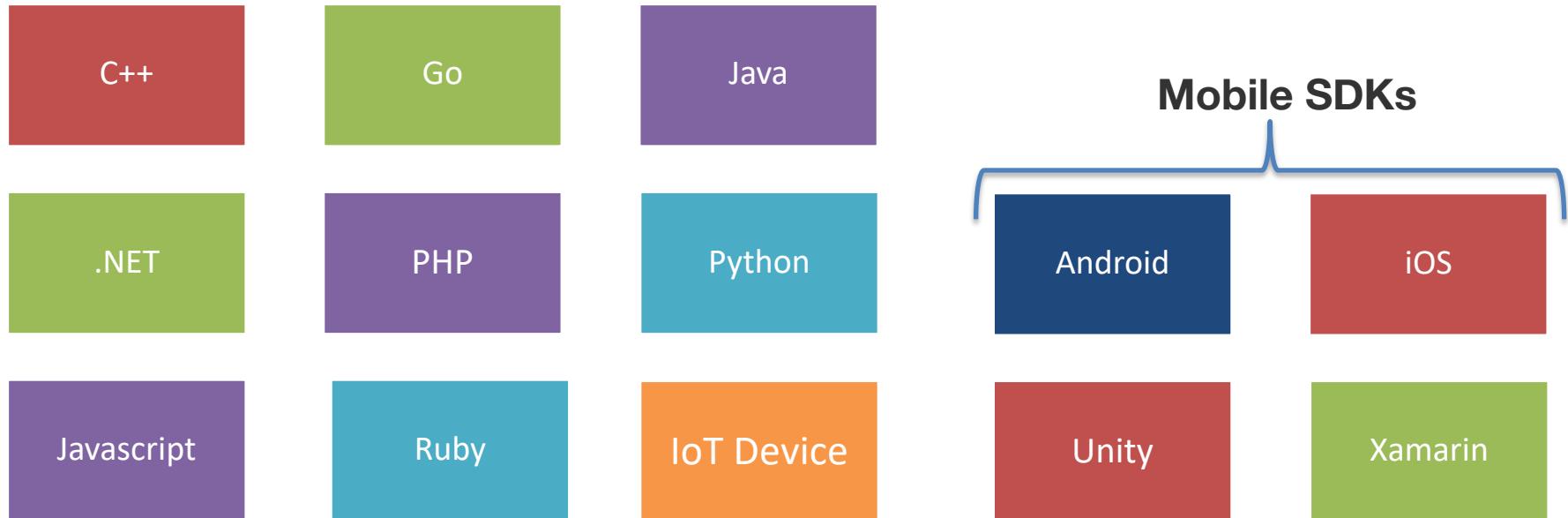
AWS Certified Developer (Associate) Crash Course

Developing in AWS

AWS SDKs

AWS SDK Overview

Amazon provides SDKs for many popular languages.



AWS SDK Overview

Amazon also provides toolkits for popular IDEs

AWS Toolkit
for Eclipse

AWS Toolkit
for Visual Studio

AWS Tools for
Visual Studio Team Services

AWS SDK Overview

Key Points

- Each SDK is developed independently.
- Not all SDKs provide access to all services.
- IAM Programmatic Users are used.

SDK Overview

Getting Help

- Forums:
<https://forums.aws.amazon.com/category.jspa?categoryID=7>
- Tools for Amazon Web Services:
Dev. Tools | SDKs | IDE Toolkits | CLI
<https://aws.amazon.com/tools/>

SDK Overview

SDKs

Simplify using AWS services in your applications with an API tailored to your programming language or platform.

Java

[Install »](#)

[Documentation »](#)

[Learn more »](#)

.NET

[Install »](#)

[Documentation »](#)

[Learn more »](#)

Node.js

[Install »](#)

[Documentation »](#)

[Learn more »](#)

PHP

[Install »](#)

[Documentation »](#)

[Learn more »](#)

Python

[Install »](#)

[Documentation »](#)

[Learn more »](#)

Ruby

[Install »](#)

[Documentation »](#)

[Learn more »](#)

Browser

[Install »](#)

[Documentation »](#)

[Learn more »](#)

Go

[Install »](#)

[Documentation »](#)

[Learn more »](#)

C++

[Install »](#)

[Documentation »](#)

[Learn more »](#)

<https://aws.amazon.com/tools/> - Scroll down to SDKs

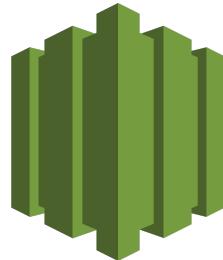


AWS Certified Developer (Associate) Crash Course

Developing in AWS

CodeStar

AWS Codestar Overview



AWS Codestar

“AWS CodeStar lets you quickly develop, build and deploy applications on AWS.”

AWS Codestar Overview

Software Project Management

- Templates for web applications, web services
- Project resources are configured to work together
- Assign project team members the roles they need to access tools and resources
 - Permissions are applied to necessary resources and services
No IAM management necessary

AWS Codestar Overview

- Single pane of glass activity monitoring:
 - Commits
 - Changes
 - Builds
 - Deployments
- Integrated development toolchain for your project
- Automated deployment after changes
- Issue tracking, JIRA integration
- Project Wiki



AWS Certified Developer (Associate) Crash Course

Developing in AWS
CodeCommit

AWS CodeCommit Overview



AWS CodeCommit

“AWS CodeCommit is a fully-managed source control service that makes it easy for companies to host secure and highly scalable private Git repositories.”

AWS CodeCommit Overview

Salient Features

- Fully managed source control solution
- Data encrypted in flight and at rest
- Highly scalable, redundant, and durable architecture
- Store any file type; no repo size limits
- Tight integration with other AWS Developer Tools, e.g. CodePipeline, CodeStar
- Works with existing Git tools and IDEs with Git support
- SNS Topic publishing upon Pull Request and Commit Comment events



AWS Certified Developer (Associate) Crash Course

Developing in AWS
CodePipeline

AWS CodePipeline Overview



AWS CodePipeline

“AWS CodePipeline is a continuous integration and continuous delivery service for fast and reliable application and infrastructure updates.”

AWS CodePipeline Overview

- Builds, tests, and deploys code after every change
- Governed by Release Process Models

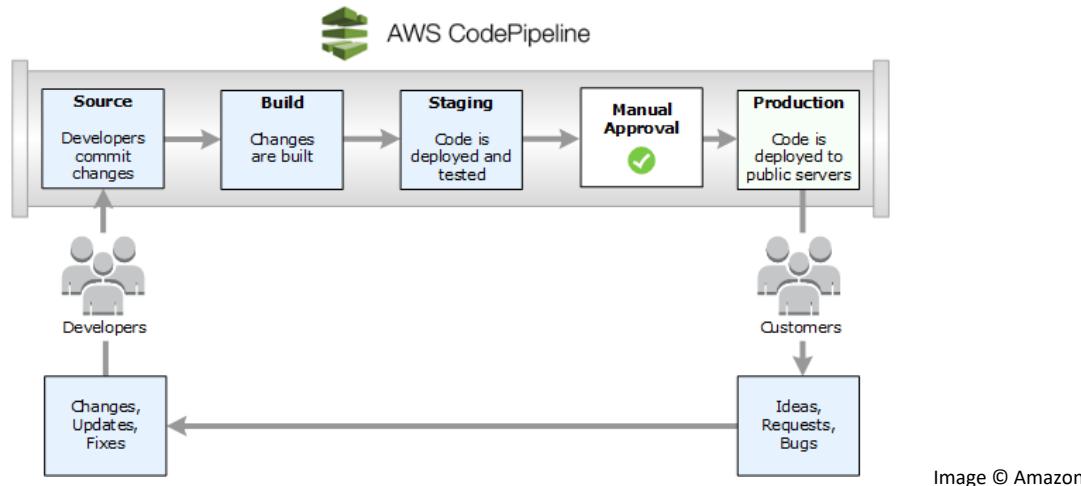


Image © Amazon

AWS CodePipeline Overview

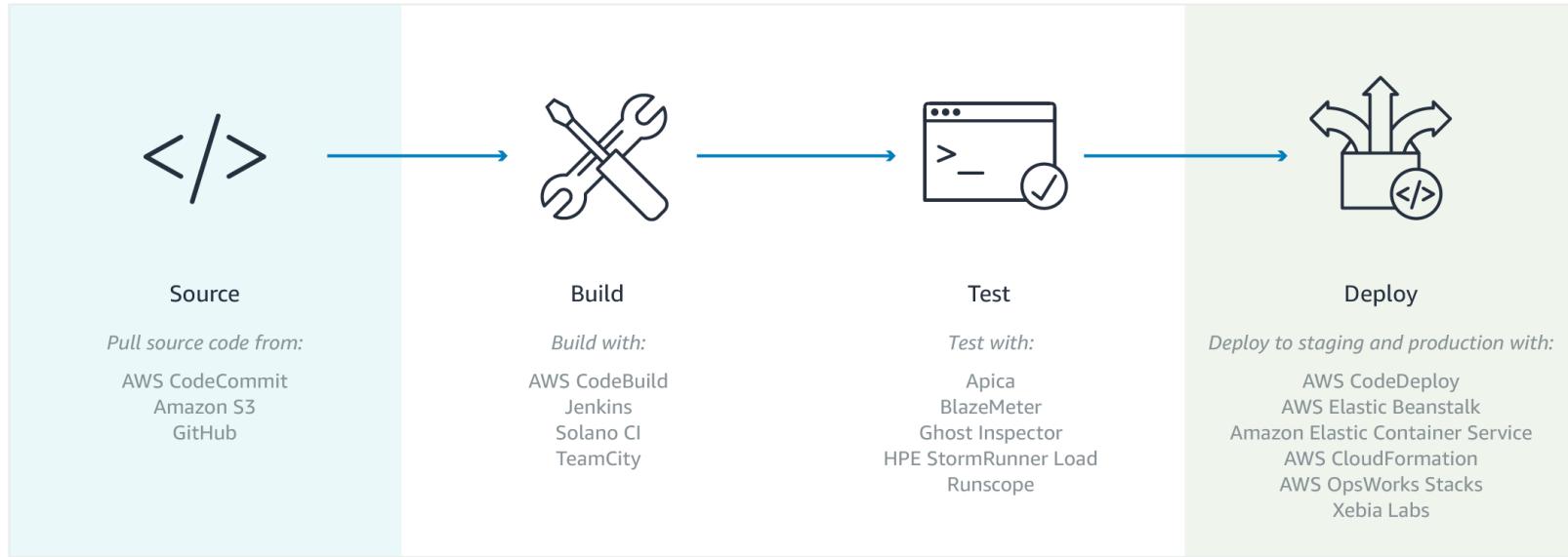


Image © Amazon

AWS CodePipeline Overview

CodePipeline Lifecycle

- Commit to repository triggers pipeline to run
Provides the output artifact from the **Source** stage
- Source artifact is ingested as an input artifact to
the **Build** stage
Output artifact from Build Stage created
Examples: application or Docker image
- **Build** artifact ingested as an input artifact to
the **Deploy** stage
 - Staging or Production environment in AWS
 - Docker -> ECS



AWS Certified Developer (Associate) Crash Course

Developing in AWS
CodeBuild

AWS CodeBuild Overview



AWS CodeBuild

“AWS CodeBuild is a fully managed build service that compiles source code, runs tests, and produces software packages that are ready to deploy.”

AWS CodeBuild Overview

- Managed build servers
- On-demand scaling
- Preconfigured build environments for popular languages

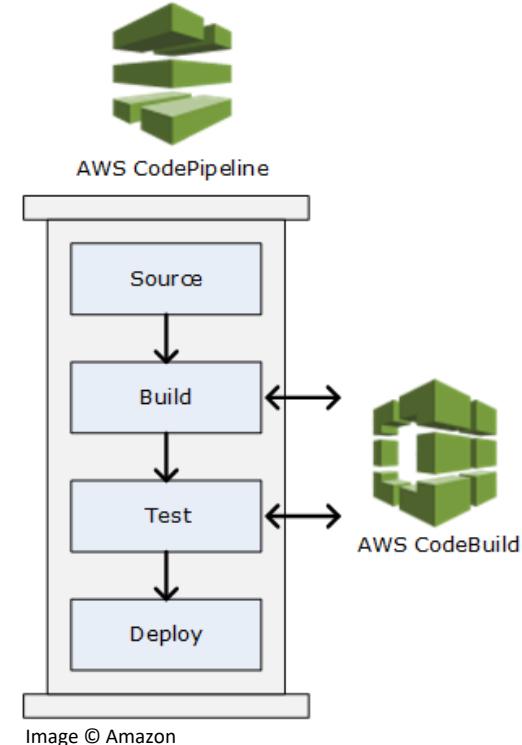


Image © Amazon

AWS CodeBuild Overview

Source Providers:

- S3
- AWS CodeCommit
- Bitbucket
- GitHub
- GitHub Enterprise

Operating Systems:

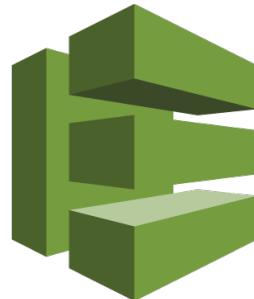
- Windows
- Ubuntu



AWS Certified Developer (Associate) Crash Course

Developing in AWS
CodeDeploy

AWS CodeDeploy Overview



AWS CodeDeploy

“AWS CodeDeploy is a service that automates software deployments to a variety of compute services including Amazon EC2, AWS Lambda, and instances running on-premises.”

AWS CodeDeploy Overview

Salient Features

- Automated deployments
- Maximize availability during deployment.
- Incremental change introduction
- Application health tracking
- Halt and rollback on error
- Centralized control
- Detailed reporting
- Push notifications of live updates regarding deployments

AWS CodeDeploy Components

AWS CodeDeploy Component	EC2/On-Premises	AWS Lambda
Deployment group	Deploys a set of instances to which a new revision is deployed.	Deploys a Lambda function version on a high-availability compute infrastructure.
Deployment	Deploys a new revision that consists of an application and AppSpec file. The AppSpec specifies how to deploy the application to the instances in a deployment group.	Deploys a new revision that consists of an AppSpec file. The AppSpec specifies which Lambda function version to deploy.
Deployment configuration	Settings that determine the deployment speed and the minimum number of instances that must be healthy at any point during a deployment.	Settings that determine how traffic is shifted to the updated Lambda function versions.
Revision	A combination of an AppSpec file and application files, such as executables, configuration files, and so on.	An AppSpec file that specifies which Lambda functions to deploy and update.
Application	A collection of deployment groups and revisions. An EC2/On-Premises application uses the EC2/On-Premises compute platform.	A collection of revisions. A Lambda application uses the AWS Lambda compute platform.

AWS CodeDeploy Overview

Deployment Destinations

- **EC2/On-Premises:** Deploy updated versions of software to servers that can be Amazon EC2 cloud instances, on-premises servers, or both.
- **AWS Lambda:** Deploy updated versions of Lambda functions.

AWS CodeDeploy Overview

Components

AWS CodeDeploy Component	EC2/On-Premises	AWS Lambda
Deployment group	Deploys a set of instances to which a new revision is deployed.	Deploys a Lambda function version on a high-availability compute infrastructure.
Deployment	Deploys a new revision that consists of an application and AppSpec file. The AppSpec specifies how to deploy the application to the instances in a deployment group.	Deploys a new revision that consists of an AppSpec file. The AppSpec specifies which Lambda function version to deploy.
Deployment configuration	Settings that determine the deployment speed and the minimum number of instances that must be healthy at any point during a deployment.	Settings that determine how traffic is shifted to the updated Lambda function versions.
Revision	A combination of an AppSpec file and application files, such as executables, configuration files, and so on.	An AppSpec file that specifies which Lambda functions to deploy and update.
Application	A collection of deployment groups and revisions. An EC2/On-Premises application uses the EC2/On-Premises compute platform.	A collection of revisions. A Lambda application uses the AWS Lambda compute platform.

AWS CodeDeploy

Deployment Type Options

- **In-place deployment:**

Application stopped, new version installed, new version started and validated. Load balancers can be used to remove and add instances from service during the upgrade.

- **Blue/green deployment:**

EC2/On-Premises: New instances launched, test if needed, shift LB targets.

Lambda: API calls are moved to a new serverless environment with the new function.

Developer Tools Overview

Developer Tools Demo

SEGMENT 7



AWS Certified Developer (Associate) Crash Course

Test Preparation

Exam Format

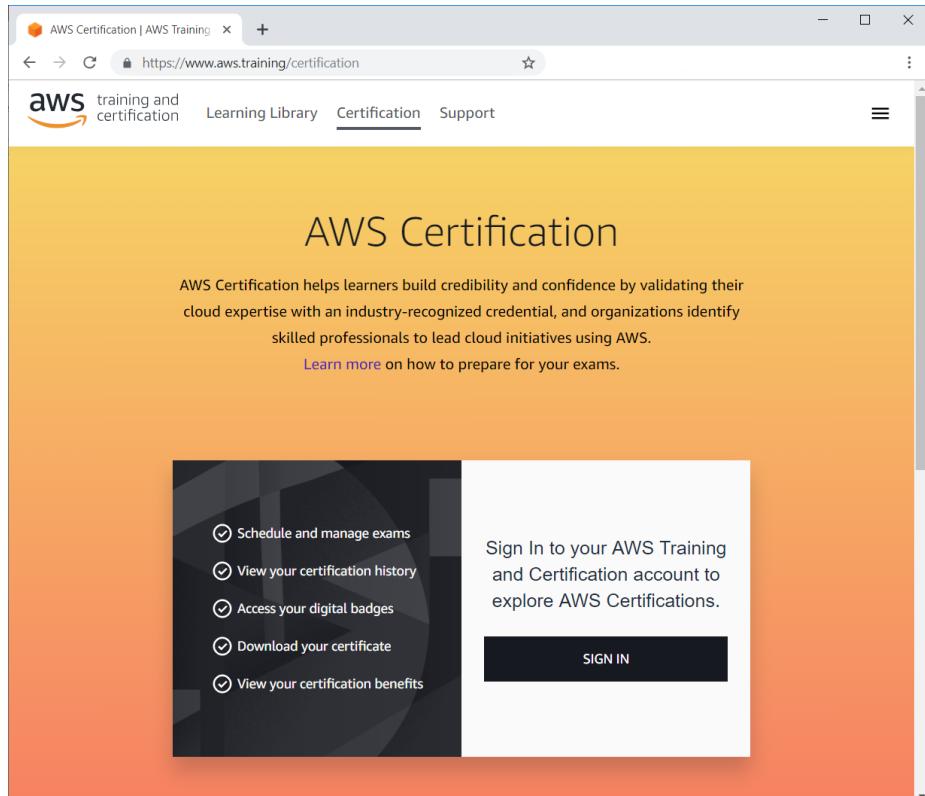
- 80 Minutes
- 55 Questions
- Standard delivery format,
proctored.
- PSI or Pearson VUE

Exam FAQ

- Retakes: 14 days
- Instant scoring upon completion.
- Beta exams – 90 days before go-live.
- 24 Hour cancellation policy, no penalty.

Exam Registration

- <https://aws.training/certification>
- Create Account
- Name must match ID!
- Must schedule exam through aws.training.
- \$150 USD
- Where to register



Test Locations / Types

- Pearson VUE, added April 2019.
 - 5,000 testing centers, 180 countries
- PSI
 - Kiosk Option

Whitepapers

aws.amazon.com/whitepapers

- AWS Security Best Practices whitepaper
August 2016
- AWS Well-Architected Framework whitepaper
November 2017
- Architecting for the Cloud AWS Best Practices
February 2016
- Practicing Continuous Integration and Continuous Delivery on AWS Accelerating Software Delivery with DevOps whitepaper
June 2017

Whitepapers

- Microservices on AWS whitepaper
September 2017
- Serverless Architectures with AWS Lambda whitepaper
November 2017
- Optimizing Enterprise Economics with Serverless Architectures whitepaper
October 2017
- Running Containerized Microservices on AWS whitepaper
November 2017
- Blue/Green Deployments on AWS whitepaper
August 2016

AWS Practice Tests

- Practice exams available for all exams:
 - Foundational
 - Associate
 - Professional
 - Specialty
- Online, Timed
- Experience the exam format

AWS Developer Associate Blueprint

<https://aws.amazon.com/certification/certified-developer-associate/>

Recommended AWS Knowledge (for the test)

- **One or more years of hands-on experience** developing and maintaining an AWS based application
- In-depth knowledge of at least one high-level programming language
- Understanding of core AWS services, uses, and basic AWS architecture best practices

AWS Developer Associate

Recommended AWS Knowledge – Continued

- Proficiency in developing, deploying, and debugging cloud-based applications using AWS
- Ability to use the AWS service APIs, AWS CLI, and SDKs to write applications
- Ability to identify key features of AWS services
- **Understanding of the AWS shared responsibility model**

AWS Developer Associate

Recommended AWS Knowledge – Continued

- Understanding of application lifecycle management
- Ability to use a CI/CD pipeline to deploy applications on AWS
- Ability to use or interact with AWS services
- Ability to apply a basic understanding of cloud-native applications to write code
- Ability to write code using AWS security best practices (e.g., not using secret and access keys in the code, instead using IAM roles)

AWS Developer Associate

Recommended AWS Knowledge – Continued

- Proficiency writing code for serverless applications
- Understanding of the use of containers in the development process



Developer - Associate

Good Luck!